Chapter 1

Introduction

1.1. What is Computer Assisted Instruction?

A Computer Assisted Instruction (CAI) system is an instructional system in which instructional content or activities are delivered via computer (Hannafin and Peck, 1988). CAI systems are a type of Computer Assisted Learning (CAL) systems. There are three major types of CAL systems:

- Computer Assisted Instruction (CAI) systems where the computer actually teaches the student skills;
- Computer Managed Instruction (CMI) systems where the computer keeps a record of the student's progress, assigns lessons and administers tests but it is not involved in teaching the student; and
- Diagnostic/Prescriptive Applications where the computer is used to administer and evaluate tests, and recommends future teaching material.

CAI systems have evolved considerably since their inception in the 1950s. CAI systems range from systems that are unable to provide feedback or individual instruction to students, to Intelligent Computer Assisted Instruction (ICAI) systems or as they are more usually termed Intelligent Tutoring Systems (ITS) (Nwana, 1990). ITS make use of cognitive science and artificial intelligence techniques.

Typically, a CAI system can be used to do any or all of the following:

- teach a particular subject;
- collect detailed performance data to better understand student problem solving techniques; and
- provide individualised instruction and the opportunity to work in group situations.

Some of the research issues associated with the design and construction of a CAI system are as follows:

- realising different teaching strategies through different teaching tactics;
- organisation of the teaching material, and the method of presenting the material to the student;
- student monitoring to determine the steps in the instructional process and to collect helpful information on the student's performance to evaluate learning issues;
- matching student answers with expert answers; and
- the Human-Computer Interface, in particular the reduction of the information and communication gap between the user and the CAI system.

1.2. Research Aims and Methodologies

The *objective* of this thesis is to investigate the research issues outlined above for the design and implementation of an extendible CAI system, and to undertake some case studies and comparative studies to demonstrate and evaluate its capabilities.

To achieve this objective the following *research aims* have been set in this thesis:

- identify suitable teaching strategy that is appropriate for teaching LOGO programming concepts and Geometry concepts;
- identify techniques of organising the instructional material and techniques of presenting these materials to the students for effective learning;
- develop student monitoring mechanisms to enable student's performance data collection;
- develop effective techniques to match student answers with expert solutions for a particular task; and
- develop a suitable human-computer interface to maximise information flow and to minimise the communication gap between the user and the system.

The *development methodology* used in implementing the CAI is known as "propose, evaluate and revise". This is an iterative development method, where each iteration attempts to improve on the previous solution to a particular problem. As an example, the application of this development method in organising and presenting instructional material, developing expert solutions and developing human-computer interface methods were firstly proposed by the author and then tested by students in the target age group and their teachers. Following evaluation, revisions were made and the CAI system was retested. This cycle was iterated until the students and teachers found the CAI system to be totally satisfactory. In the case of the student monitoring component the author evaluated and revised the CAI system until sufficient data was collected to gain the required information regarding students learning abilities.

1.3. Research Outcomes

The main research outcomes are:

- development of an extendible CAI system to enable effective teaching of the subjects LOGO and Geometry, and
- using the CAI system to collect appropriate students' performance data during the learning process to better understand the learning abilities of the students, and the common errors these students make during their learning process.

The CAI system has the following capabilities:

- highly flexible design where the system can be used for a range of different age groups. It can be used for teaching or class projects, and it can be used by teachers to add to or alter existing teaching materials;
- ability to handle high fault tolerance in dealing with user responses;
- high level of student interaction with limited control of lesson sequencing;
- suitable for use by individual students or small groups of students;
- ability to record relevant data that could be used by the teacher to study the learning process of the students;
- ability to accept a variety of correct solutions to the same task; and
- to enable the students to learn by free exploration once they have completed learning LOGO and Geometry.

The CAI system also has the following features:

- uniform and flexible structure for presentation of teaching material;
- uniform screen layout; and
- the system does not require additional media such as books and manuals to help the student during the learning process.

The overall achievements outlined in this section demonstrates the successful achievement of the objective of this thesis (c.f., Section 1.2).

1.4. Organisation of the Thesis

This thesis is organised in the following manner:

Chapter 2 contains a literature survey which examines the different theories of learning, and identifies tutorial and discovery learning as the two most appropriate teaching strategies for teaching LOGO and Geometry, and justifies CAI as an appropriate medium for their realisation.

Chapter 3 examines the requirements of a CAI system used for teaching LOGO and Geometry. An overview is given of the conceptual design of the system, and the author describes the connection between the modules in the CAI package. The design of the system is evaluated to determine that the requirements are met.

Chapter 4 describes the Lesson Component in detail. This is the teaching component of the system. The author looks at the types of knowledge needed to teach the subjects successfully, and the methods used to represent this knowledge. This chapter outlines the "characteristics" of good lesson design and details the algorithms used. Chapter 4 also investigates the Expert Solutions Editing module which is used by the teacher to enter the solutions to the tasks set.

Chapter 5 describes the Student Monitoring Component. This component is used by the CAI system to adjust the sequencing of instruction. Student monitoring also supplies the teacher and researcher with the data required to build a model of student learning which helps the teacher to understand how the students learn, and improve the instructional process accordingly. The author outlines the design of the Student Monitoring Component and describes the data collected.

Chapter 6 discusses the Human-Computer Interface which controls the interaction between the user and the system. The user may be a student, group of students or a teacher. The author discusses the factors that must be considered for these two different types of users of the CAI system, and describes the design of the student-system interface and the teacher-system interface.

Chapter 7 outlines the measures taken to ensure the CAI system was ready to be used in the classroom environment (i.e., operationally ready). Firstly, the factors that determine whether a CAI system is operationally ready is explained. Secondly, the evaluation of the CAI system for operational readiness is summarised.

Chapter 8 describes the first of the two case studies undertaken. This case study investigated the level of success of students learning LOGO programming concepts, and whether there was any correlation between this and the students methods of problem solving.

Chapter 9 outlines the second of the two case studies. Students were taught elementary Geometrical concepts. The case study compared students with known LOGO and Geometry skills working firstly in randomly allocated groups, and secondly in groups of their own choosing.

Chapter 10 compares the system's capabilities against a commercially developed CAI system, using the following criteria:

- flexibility of the system;
- quality of directions;
- quality of screen design;
- quality of responses;
- level of student interaction and control;
- self-contained or need for external information;
- suitable for use by individual students or small groups;
- collection of student performance data; and
- quality of testing student performance.

This chapter also compares the lesson design in the system with the lesson design in the LOGO school curriculum used by the Chapel Hill and Carrboro City Schools, USA.

Chapter 11 reviews the objectives of this thesis and concludes that these objectives have been achieved. The author then outlines the limitations of the current CAI system and discusses future work to enhance the system.

Chapter 2

Literature Survey

2.1. Introduction

Computational based instructional systems are increasingly being used in industry and education in recent years. Over the past centuries, educationists have developed numerous teaching strategies and related tactics that could be effectively used by teachers. Some strategies have been in existence for quite some time such as drill and practice, and discovery learning, while others are being developed. Thorndike (1969) argued "that bonds between stimuli and responses are strengthened through exercise in which success is recorded" which is the philosophy that drill and practice is based upon. In the 1960s a number of educators encouraged a discovery approach to learning (Gadanidis, 1994). They argued that learning a subject should be a process that students should experience, and that students should search for and subsequently discover patterns and unify structures. Some of the other contemporary teaching strategies include Game Playing, Simulation, and Tutoring. All these teaching strategies can be used as a basis for the teaching strategy in a CAI system. The choice of strategy depends on the subjects being taught and the aims of the instructional system.

In this chapter, the author will review the contemporary instructional paradigms, teaching strategies and associated teaching tactics for computational instructional systems. To achieve these, the author has organised this thesis in the following manner. In Section 2.2, the author will investigate the following instructional paradigms: behaviourism, systems theory, cognitive theory and constructivism. Following this, in Section 2.3 and Section 2.4 the author will explore contemporary teaching strategies and the related teaching tactics, respectively. In Section 2.5 the author will describe the subject area (LOGO instruction and Geometry) for which one or more of the teaching strategies were implemented. In Section 2.6, a review on CAI is carried out in an attempt to demonstrate how the teaching strategies can be implemented in such a software system successfully. In Section 2.7, the author justifies the use of micro-computer and CAI as a medium to realise the teaching strategies which are used to carry out appropriate instruction on the subject area to educate the students (users of this system). Finally, Section 2.8 provides a summary of this chapter.

2.2. Instructional Paradigms

Four theories have been the basis for investigating the impact of computers in the teaching and learning process. They are behaviourism, systems theory, cognitive theory, and constructivism. Some of the experimentations of these theories are outlined in the following subsections.

2.2.1. Behaviourism

Of the theories supporting the use of computers in education, behaviourism has had the most impact. Behaviourism is based on the principle that instruction should be designed to produce observable and measurable behaviours in the student. After completing a lesson, students should be able to do something that they could not do, or could not do as well, before the lesson. Many educators have found this technique effective (Simonson and Thompson, 1990). Behavioural objectives are easy to develop and have been shown to be related to improvement in student achievement. Behaviourism may be best explained by looking at the work of the behaviourists Thorndike (1969), Pavlov (1940) and Skinner (1969).

Thorndike (1969) wrote about psychology and education in the early part of this century. His work was so influential that his ideas dominated thinking in both psychology and education for over 50 years (Simonson and Thompson, 1990), although Thorndike's views were challenged by many prewar and postwar progressive educators (Gaganidis, 1994). Thorndike (1969) stated that learning was based on a series of connections between the problems of a particular situation and what had been accomplished in the past. The Law of Effect is the main contribution of Thorndike's connectionism. It states that when a modifiable connection between a situation and a response is made and is accompanied or followed by a satisfying state of affairs, then the connection's strength is increased (Simonson and Thompson, 1990). Connectionism advocates that teachers break down complex tasks into simple tasks, and positive reinforcement of desirable outcomes will build up connections between the instructional situation and the required behaviour. The establishment of specific goals of teaching, the expectation that goal-related changes could be measured, and the idea that large tasks should be subdivided into simpler ones became basic concepts of behaviourist thought (Simonson and Thompson, 1990).

Skinner's work was based on Thorndike. Skinner (1969) maintained that there were two types of learning: classical conditioning; and operant conditioning. The first was Pavlov's classical conditioning, where a stimulus was applied to an organism to produce a response. Pavlov's experiments were with dogs. He observed that dogs that were about to be fed began to salivate (Pavlov, 1940). Pavlov would ring a bell before food was placed in their mouths, and after a short time the dogs began to salivate when the bell was rung, even if they did not receive food. Learning occurred when the new stimulus produced the response, even though the original stimulus was removed.

The second type of learning that Skinner believed existed is called operant conditioning. Operant conditioning encourages the use of reinforcement following desirable actions to promote wanted changes in behaviour

Modern approaches to teaching include a third type of learning, namely collaborative learning (Adams and Hamm, 1990). In this learning environment students work together in groups, discuss topics and take charge of their own learning. Students learn techniques for analysing, interpreting, negotiating and communicating their information as a team.

In summary, Skinner's contributions to educational practice include the following techniques (Simonson and Thompson. 1990):

- stating objectives in terms of desired terminal behaviour;
- assessing a student's previously acquired behaviours before any instruction, to determine if they are ready for a lesson;
- placing a student in a sequence of instruction where he/she can achieve at the 90% level;
- using teaching machines to reinforce and to strengthen desirable behaviours; and
- recording a learner's progress through a lesson to gain feedback for revising the lesson.

Skinner has supported the use of computers in education because he believes that when computers are correctly programmed, they become ideal teaching machines. Small sections of information can be introduced by computer lessons, and students can be positively reinforced when successful learning takes place.

2.2.2. Systems Theory

The second theory used to examine the use of computers for teaching is systems theory. Systems theory provides a set of rational procedures for designing instructional programs. Systems theorists state that events should be studied in relationship to other events, and their impact measured. Systems theory requires controlling all but two variables: the variable being studied; and the variable that is thought to influence it. In this way the scientist can observe phenomena and comment on how they are influenced.

The set of procedures the systems approach gives instructional designers to follow are based on the ideas of objectivity and causality. Objectivity implies that a student can observe events, understand the world around them and use this information to explain the causality of events. Causality implies that events can be planned for and predicted. Systems theorists consider the world to have a natural order, and the rules that govern this natural order can be determined and used to predict what is likely to happen in the future.

Simonson and Thompson (1990) describe the systems approach for designing instruction to encompass the following three stages:

- In the first stage, the broad objective of the instructional problem is defined, and the instructional situation analysed. Information relating to the students previous experience and learning styles are collated and matched to the instructional resources and implementation techniques. This ensures that input is correctly processed by the system. The procedures used to manage the instructional activities are organised.

- In the second stage specific behavioural objectives are specified, teaching methods are identified, materials are chosen or developed, and the instructional programs are designed.

- In stage three the instructional materials and techniques are monitored. The feedback is used to alter the functioning of the system until the new instructional system is determined to be totally effective.

Consequently, the systems approach incorporates ways of looking at complex organisational problems that takes into account all contingencies.

2.2.3. Cognitive Theory

Cognitive theory is the third theory which provides direction for computers in teaching. Educational psychologists and learning theorists have begun to move away from the behaviourist approach and have advocated a closer look at the internal processes that occur in learners during instruction (Simonson and Thompson, 1990). Cognitive theory centres on the way facts are processed by the brain. Proponents of cognitive theory, such as Piaget and Papert, advocate that instruction be matched to the cognitive structure of the student.

Cognitive theory gives several guidelines for instructional design. They are as follows:

- A great deal of importance is placed on the student having the inclination to learn.

- The structure and form of knowledge presented is important. Part of cognitive theory is based on the concept that children are first able to understand real life situations, then graphic representations of reality, and finally abstract symbols. Therefore, it is important to know a student's previous experience of the topic.

- The sequence of instruction must be considered. Students with different learning styles respond better to different sequencing of instructional material. Students who have the left hemisphere of the brain as the dominant hemisphere typically process data sequentially. These students tend to respond best to instruction that is very structured and in a logical, easy to follow order. Students who have the right hemisphere of the brain as the dominant hemisphere typically process data in parallel, and consequently are involved in pattern recognition. These students tend to learn more from instruction that first shows them what they are expected to learn and then fills in the details.

- The form and pacing of reinforcement is important. Learning depends on the student knowing the basic concepts before more complex concepts are taught. Feedback should be appropriate for the student's current level of knowledge, and is a mechanism for supporting correct mental functioning.

- Discovery learning is one important strategy that incorporates much of cognitive theory (Simonson and Thompson, 1990). Discovery learning involves placing the student into a situation the student can explore and discover concepts by himself/herself without direct instruction.

2.2.4. Constructivism

Constructivism is the belief that knowledge is personally constructed from internal representations by individuals using their experiences as a foundation (Jonassen, 1990). Constructivists believe that if knowledge is constructed individually, then there is no objective reality, and our own experiences determine the reality (Jonassen, 1990).

Constructivist-learning theory asserts that "all mental activity is constructive". Even in learning situations that are considered passive, such as a lecture, students construct their own understanding (Gadanidis, 1994). Thus, from the constructivist point of view, the question is not whether students construct understandings of concepts but rather how good are their constructions (Gadanidis, 1994).

Cognitive Flexibility Theory is a constructivist theory of learning and instruction that emphasises the real-world complexity and ill-structuredness of many knowledge domains (Spiro, Feltovich, Jacobson and Coulson, 1991). At advanced stages of knowledge acquisition content becomes more complex and the relationships across the cases that knowledge has to be applied become more irregular (Spiro and Jehng, 1990). That is, the student is working within a complex and ill-structured domain. If the problems related to content complexity and irregularity are ignored then this results in conceptual oversimplification and the student is not able to transfer the knowledge to other situations. Spiro et al. (1991) emphasise approaching the same items of knowledge from different points of view and for different purposes, and random access computer technologies make it a straightforward matter to revisit material in a variety of different contexts.

Constructivist cognitive theories of learning state that decontextualised learning is less meaningful and less memorable. Information is best learned through the context of some real-world problem or experience and the most effective contexts are problem or case based activities which immerse the student in the situation and require him/her to acquire skills and knowledge to solve the problem (Brown, 1993). This is the concept of Anchored Instruction. In 1991 "The Cognition and Technology Group" at Vanderbilt USA, developed video environments that provided an opportunity for exploration and collaboration. The video environments depicted real-life adventures that could be explored at many levels, and from multiple points of view. Initially the students were not shown the conclusion of each video and they were asked to solve a particular problem. Students were not shown the conclusion until after they had solved the problem. Different case based projects were given to the students using the videos and students could then determine which problem solving strategies were relevant to only one situation and which ones were generalisable.

In summary, behaviourists look at outcomes, system theorists look at the events that influence entire systems, and cognitive scientists look at students internal processing. All these approaches advocate reinforcement or feedback, and all approaches are interested in how instruction is sequenced. Behaviourists, system theorists and cognitive scientists all support individualised instruction. The computer is ideally suited, by virtue of the flexibility it can provide, for fostering cognitive flexibility an element of constructive processing (Spiro, Feltovich, Jacobson and Coulson, 1991).

2.3. Teaching Strategies

Teaching Strategies are best described as generic methods that can be utilised by educators to instruct students. Ford (1987) outlined the following categories of teaching strategies relevant to CAI:

- Drill and Practice;
- Game Playing;
- Simulation;
- Tutoring; and
- Discovery Learning.

Drill and Practice is one of the traditional educational practices that involve firstly the process of teaching a particular skill and secondly the students practising the application of this skill by carrying out a large number of exercises (Hannafin and Peck, 1988). A typical example is to teach the students the skill of arithmetic division and getting the students to practice this skill by solving sample problems.

Game Playing has been used by educators for centuries. An example of using game playing principles in education is to educate students on coordinate geometry using the game premise of searching for submarines (Hannafin and Peck, 1988). Using the game playing principles, an educator can also teach problem solving methods by asking students to assemble mechanical components to form machines capable of building particular products. Although the game playing teaching strategy can occasionally teach new information or concepts, primarily games are used to reinforce concepts taught elsewhere as in Drill and Practice. Game playing teaching strategy tends to be highly motivating.

Simulation is an educational practice that teaches simple facts about the event being simulated as well as teaching procedural logic (what to do next) and processes (principles well understood by experts in the field). Simulations are often used to model complex situations or circumstances, such as resolving political conflicts, learning to land an aircraft, and making diagnosis of diseases. Simulation provides the students with the opportunity to make decisions within an environment which:

- acts as a substitute for a real situation;
- informs the student of the consequences of each decision made; and
- avoids the costs, dangers and time constraints associated with the real situation.

While a simulation is as realistic as possible, the level of realism is limited by cost, danger and time constraint. In some instances, the level of realism in a simulation is deliberately limited to focus the student on certain aspects of the situation.

Tutoring is an educational practice where the interactions between the student and the tutor is on a one-to-one basis. This teaching strategy teaches and monitors the progress of learning. Concepts are presented, the student's understanding of each concept is checked, and subsequent instruction is provided based on the student's responses (Hannafin and Peck, 1988). This teaching strategy may include the ability to adjust the level of difficulty of the tasks set, and offering detailed explanations when required. Features such as these mean the tutoring teaching strategy accommodates individual differences among learners (Dennis and Kansky, 1984). An example of this tutoring teaching strategy can be found in the "Bodyworks" CAI system which is used for teaching anatomy.

Discovery Learning was explained by Piaget (1970). Piaget has a belief of how children learn and hence developed a philosophy of education. For Piaget, a child's mental capacities consist of simple "thought structures" which are established and tested in the course of the child's exploration of his/her environment. At a later stage of maturity, these simple thought-structures are combined to form a more complex or abstract thought structure, which in turn can form the basis for further intellectual growth. Exploration and discovery are seen from the Piagetian point of view as key elements in learning. An environment is established in the classroom where a new idea is discovered or connected to some other idea. An example of discovery learning teaching strategy is building a LOGO computer program by developing and testing simple procedures, then incorporating them into more complicated ones. The parallel between LOGO activity and thinking is a crucial element in the case for LOGO's relevance to education (Martin, 1986). Many argue that children must develop cognitive understanding, and discover things for themselves if true learning is to take place (Willis, Johnson and Dixon, 1983).

2.4. Teaching Tactics

The above teaching strategies are realised by different teaching tactics. Teaching tactics involve making decisions at each cycle of interactions as to what to do next to foster learning on the part of the student (Ford, 1987).

The teaching tactic used by the *drill and practice* strategy is to repeat instruction until the student understands the material. There is no new information taught as the actual teaching of the concept is provided prior to the drill and practice session. A drill and practice activity serves to fix some association or to refine some skill. It is not the function of drill and practice to motivate, explain, justify, or expand upon an idea (Dennis and Kansky, 1984). The teaching strategy is concerned with:

- choosing the type of question (e.g., multiple choice, short-answer);
- setting the restrictions on the student's input (e.g., does spelling count?);
- providing help through extra prompts;
- providing immediate and specific feedback;
- recycling questions; and
- determining when the session should end.

The teaching tactic used by the *game playing* strategy to foster student learning is to present attractive images, and to hold the student's attention by presenting the subject material in a creative form. The teaching tactic adopted by the instructional game playing strategy primarily follows the idea that one can never be passive when involved in game playing because game playing is individualised. What we get from it depends upon the prior knowledge we bring to it. Play encourages the formation and testing of strategies, the examination of guiding principles (rules), and the exploration of the effects of modifying the guiding principles (Dennis and Kansky, 1984). The teaching tactic employed by the game playing strategy incorporates:

- the concept of an opponent (the opponent may be another person, a computer, or even the task itself);
- given rules and goals of play which define the game;
- feedback on the student's progress; and

- the concept that the student's success is related to his/her own actions.

In *simulation* the teaching technique used is to accept the student's input and respond as the system being simulated would respond. This teaching tactic encourages the student to experiment with different decisions, and to think through the consequences of his/her actions. The teaching tactic employed in the simulation teaching strategy are as follows (Dennis and Kansky, 1984):

- the opening scene (scenario) details the student's role in the simulation, states a task or goal, and reveals any limitations imposed upon time, assistance, or related resources ;
- decision pairs which consist of a decision and the consequence of that decision;
- the sequence of decision pairs which lead to the successful completion of the tasks (solution paths);
- strategy initiation sections which are instigated every time all the consequences of the student's previous strategy have been completed; and
- transition messages which inform the student of the mechanical steps involved in the simulation.

The decision pairs involved in a simulation can be rated on a scale of good, bad and those which lie somewhere between. The good decisions help the student reach a solution of the task, and the bad decisions lead to failure. The solution paths which only include good decision pairs are known as optimal solution paths.

Tutoring differs from the other teaching strategies in that its primary concern is to teach the student new information. The teaching tactic used by this strategy is to present new concepts to the student only when previous concepts have been tested and appear to have been learned, and to use a variety of presentational styles. Remediation is given when required. The teaching tactic in tutoring teaching strategy involves:

- being able to detect when a concept has been grasped by a student, and to change direction accordingly;
- reinforcing correct answers;
- accepting a variety of correct answers;
- giving appropriate feedback when errors are made; and
- allowing the student to interrupt a lesson, and stop when he/she wishes (particularly in a CAI environment).

Discovery Learning uses the teaching tactic of placing the student in control, and enabling the student to discover his/her own solutions to problems. The students are taught how to learn. Instead of learning by example, the students are taught how to discover for themselves. The students are provided with an activity or placed in an environment where they can explore, hypothesise, conjecture and test ideas. As discoveries are made they are usually recorded and discussed to clarify the ideas before they are applied.

2.5. LOGO and Geometry

LOGO is one of a number of computer languages to have been developed in the field of Artificial Intelligence (Martin, 1986). Artificial Intelligence is the study of how to make computers do tasks which if carried out by people would require intelligence. LOGO was primarily meant to be a children's computer language.

The first version of the computer language that was to become LOGO was created in 1967 at the research laboratory of Bolt, Beranek, and Newman Inc., in Boston. LOGO was intended as a language for teaching mathematical concepts to children through computer programming, and commenced as an easy-to-use version of LISP. The designers of LOGO included Seymour Papert, Wallace Feurzeig, Cynthia Soloman and Daniel Bobrow. It was Papert (a mathematics professor at the Massachusetts Institute of Technology) who first introduced LOGO into the world of education. Papert spent 12 years leading a team investigating how children learn and how computers could help them. The results were published in the book "Mindstorms: Children, Computers and Powerful Ideas" in 1980 (Papert, 1980).

LOGO programs are built up through the use of procedures, which are lists of instructions. An instruction consists of one or more LOGO commands. Simple procedures can themselves be used as single commands in more complex procedures. A computer program in LOGO is simply a collection of procedures which achieve a particular objective. For example, when we have created the procedure named SQUARE to draw a square, we can get LOGO to draw a square by merely typing in the procedure name.

When Papert created the turtle the computer became accessible to younger children and allowed them to explore shapes. Initially, the turtle was a small floor robot which children could use to draw with on a large sheet of paper, and their drawings were then replicated on the computer. Now the turtle is often represented by a small triangle, an arrow head or even a small picture of a turtle on the computer screen. The children move the turtle across the computer screen by means of commands such as LEFT, RIGHT, FORWARD, and BACK. As the turtle moves backwards and forwards it draws a line. For example, FORWARD 100 LEFT 90 will draw a line 100 units in length and then turn the turtle 90 degrees to the left.

Much research on LOGO was done throughout the 1970's mainly in the USA and Scotland. A great deal of this research was concerned with the use of LOGO in teaching mathematics. There are many ways that LOGO can be incorporated into the teaching/learning process, such as teaching word processing and teaching music, but these are not the strengths of the language. The most effective uses of the language are in the areas of mathematics, problem solving, and computer science (Burnett and Friesen, 1985). While writing a LOGO program the student assimilates information in mathematics, programming and logical deduction.

There are specific and interesting aspects about the relationship between mathematics and computers. The increase in technology has meant that the need for mathematical knowledge in everyday life has decreased (Noss, 1990). The mathematics is embedded in the technology. Noss (1990) also notes that the ideologies behind a programming language are related to the purpose for which the language was designed. LOGO is a language developed and used for mathematics. This is clearly seen by examining the way in which the elements of Geometry have been incorporated into the turtle's behaviour. LOGO is mathematical because of the way the student can access the mathematics embedded in the language. The student is also able to add to the mathematics by writing his/her own LOGO procedures. LOGO is also mathematical because students can express themselves mathematically with the language.

Hoyles (in Williams, 1990) identified three criteria for teaching mathematics in a school environment:

- the environment should generate extended student projects;
- the environment should encourage discussion and reflective experimentation; and
- the environment should illuminate student meanings and interpretations.

The closest programming environment to encompass the above criteria is the language LOGO. Following are brief descriptions of two projects undertaken using LOGO to teach mathematical concepts.

The Chiltern LOGO Project in England was set up by the Government's Microelectronics Education Program in 1982. The aim of the project was to examine the way in which 8 to 11 year old children learned to program in LOGO, in particular the potential of LOGO for learning mathematics (Noss, 1990). The findings of this project suggested that the children encountered the powerful ideas of LOGO in three related contexts:

- the syntax and semantics of LOGO (recursion, iteration, subprocedures etc.) which form the fundamental building-blocks of the language itself;
- the conceptional ideas embedded within the content of the relevant microworld. In a turtle-geometric context, these include the idea of a turtle "state" and the various turtle-theorems; in a broader context, we would include ideas such as variable and function; and
- the heuristics of LOGO, such as debugging, breaking down problems into sub-problems etc.

These categories are far from distinct. For example the computational idea of a subprocedure connects with the heuristic of breaking down a mathematical problem into more accessible parts. Similarly, the idea of LOGO "inputs" to procedures may provide a powerful computational metaphor for the idea of a mathematical function.

In a LOGO project undertaken by Williams (1990), a class of 14 and 15 year students worked with a simple LOGO procedure for producing polygons. This procedure had one variable, namely the number of sides of the polygon. The students were asked to:

- produce 3 and more sided polygons;
- experiment with non-integral numbers - therefore producing stars; and
- prepare a research proposal on topics such as finding the number of different 19 pointed starts, explaining why there were more 11 pointed starts than 12 pointed stars, and explaining what happened when negative numbers were used.

Polygons and the LOGO procedures that produce them are rich areas for mathematical learning and have become an integral part of the LOGO/mathematics education curriculum (Williams, 1990).

The students noticed that when the turtle drew a polygon it always finished up in the same direction as it started, that is, the turtle turned 360 degrees. From this, the students understood the relationship between the number of sides of the polygon and the size of the turns (external angles) required to draw the polygon.

Using LOGO in the above manner, Williams (1990) concluded the following:

- it provided a source of mathematical experimentation at several levels to students with a wide range of abilities;
- it encouraged mathematical analysis; and
- it provided a visual illustration of abstract number concepts.

2.6. Computer Assisted Instruction (CAI) Systems

Computer Assisted Instruction (CAI) system is an instructional software in which instructional content or activities are delivered via computer (Hannafin and Peck, 1988). Poirot and Norris (1987) note that, in the past, three factors have hindered the use of CAI:

- Computer Power: Logic programming problems require speed, power and memory of 16 and 32-bit personal computers.

- Research: In the past there was only a small number of researchers concentrating on CAI (Kearsley, 1987). Now research is being funded by governments, military and commercial organisations, and training and human factors specialists are interested in this area.

- Learning Theory: Advances in cognitive science are supplying a theoretical basis for designing effective instructional software - as there are models of how various cognitive tasks are performed.

With the improvement in the presentation of material, the amount of user-system interaction and the intelligence of the software, the application of CAI is made possible.

A CAI system that uses the *drill and practice* teaching strategy assumes that the student has already been taught the subject, and it only provides the means to practice his/her newly acquired skills. The Drill and Practice strategy is merely automating skills. This method is considered out-dated in many quarters. Many of the computers used in schools are still used in this manner, and can help the "slow learner". This is due not only to the patience of a machine working at a student's own pace, but to the individualisation of instruction offered. Drill and Practice programs have the potential to record a student's performance, and provide extra training in areas where the student is experiencing difficulties.

A CAI system that uses *game playing* encourages the student to develop strategies. There are two issues that must be dealt with when using computerised instructional games. The first of these issues is the fact that the use of instructional games means that a computer must be used by an individual student or a pair of students for extended periods of time before they will develop the desired strategies. This is a problem if computer resources are limited. The second issue is that while games are motivating and exciting, students seldom remember the instructional objective of the game. Games must be components of an overall instructional plan which includes organising activities preceding the games, and summarising and relating activities following the game.

A CAI system that uses *simulation* must be able to do the following:

- Address a learning objective that is relevant for the age of the students. For example, it is not meaningful to teach kindergarten children how to use a credit card.
- Present a reasonable method of teaching the subject. The question that must be asked is "do students have to engage in problem solving aspects", or "do students have to learn to handle the actual medium".
- Offer an improvement in instruction. The simulation must be as effective as the existing experience in meeting the main learning objectives and any additional learning objectives.
- Be able to be used properly given any existing time and equipment constraints.

CAI that uses the *tutorial* teaching strategy dates back to the 1960's. Linear programs were the first type of program used, and had three discernible components (O'Shea and Self, 1983):

- Program Output : The introduction of a small amount of information.
- Student Input : A response from the learner, and an immediate checking of this response to find out if it was correct.
- Program Reaction : Regardless of the response, the program moved to the next predetermined presentation of material.

Finally, a CAI that uses *discovery learning* encourages the student to explore situations and think out problems by himself/herself. Exploration of an area of knowledge or technique inevitably involves exploring paths which may turn out to have dead ends, and using materials whose properties are little known and unpredictable. Seeing where the paths go, and trying out the materials all add to the student's knowledge and understanding. The student has to think about why unexpected results may occur.

When CAI is compared to traditional forms of instruction the most common result is that there is no significant difference (Clark, 1984), (Dence, 1980), (Leiblum, 1982). This is especially true if one takes into account that in many cases CAI lessons have been prepared by skilled instructional designers whereas existing forms of instruction may not benefit from the same careful development (Bright, 1983); (Clark, 1984). Hannafin and Peck (1988) noted that although CAI has not proven to be an intrinsically superior educational medium, the following points have been substantiated:

- CAI appears to be an effective means of achieving educational objectives, both as the foremost means of instruction and as an additional form of instruction;
- when CAI is compared to other means of instruction that do not account for the individual differences between students, CAI will foster the same amount of learning but in a shorter time;
- retention of information is at least as good as retention following more traditional methods of instruction;
- students favour well designed CAI programs but reject poor programs;
- recently developed CAI lessons have been found to be more effective than earlier lessons; and
- the capability to provide immediate feedback is a key factor in CAI's efficiency and effectiveness (Caldwell, 1980).

CAI has several advantages when compared with traditional forms of instruction. Some of their advantages are listed below:

- The student's progress through a lesson is directly related to the way in which a student responds. If the student responds quickly then the CAI system moves on to the next topic. The student can take as long as he/she wishes to solve a particular problem. The student sets the pace of the instruction.

- The student's level of understanding is constantly being monitored. If the student deviates from the topic, requires extra help or needs remediation the system reacts accordingly. A CAI system caters to the needs of an individual student.

- In some CAI systems the student is in control of the instructional process, and many students feel more comfortable making mistakes on a machine (Hannafin and Peck, 1988). CAI creates a motivational environment.

- A CAI system is able to provide immediate feedback.

- It is the student who determines whether or not to continue with the lessons, and whether or not he/she wishes additional help. CAI gives control to the learner in regard to the sequence of instruction.

- CAI makes it possible to have a class of students studying different subjects simultaneously. CAI also makes it possible for instruction to be given to students in remote areas. CAI has administrative benefits.

- A CAI system may keep a detailed record of each student as the student progresses through the lessons. Thus, CAI assists the teacher with record keeping.

- CAI provides a consistent method of instruction, and can be used to verify whether or not instruction and/or learning actually took place. CAI provides lesson integrity.

2.7. Why CAI System?

The introduction of computers in homes and schools created opportunities for new ways of learning and improving the quality of learning. Microcomputers are a successful tool to aid learning in primary and secondary schools (Blemings, 1985). Children are curious and open to new ideas, and have not developed any fear or prejudices against computers. Teachers have noticed that children using computers have a greater interest in the material being presented and that their interest is maintained for longer periods of time than when using more conventional methods of teaching (Williams, 1984). The author has adopted the CAI approach for building the instructional systems for the following reasons:

- individualisation of the rate of instruction;
- individualisation of the sequence of instruction;
- specific goals can be established, and large tasks subdivided;
- good representation of the subject area (LOGO and Geometry);
- high level of student interaction;
- opportunity for group interaction and peer teaching;

- consistent evaluation of the performance of the students;
- student feedback can be given for positive reinforcement;
- unobtrusive monitoring of a student's progress;
- feedback can be given to teacher regarding students needs, and improvements to instruction;
- motivation to learn; and
- ability to use Discovery Learning Teaching Strategy.

Not everyone is enthusiastic about the use of computers in the classroom (Marcure, 1987). There are the following obstacles to be overcome:

- High quality hardware and software are expensive and consequently they are not widely available in schools. This raises the objection that the use of computers increases the gap between the financially better off schools and the poorer ones.

- There are the "normal" problems associated with learning and accepting a new technology. There are some teachers who find it difficult to adapt to the use of computers. This may be overcome through developing programs in consultation with teachers.

- Students are being introduced to computers outside of the school environment and it is a matter of concern to some educationalists that this makes it awkward to set and apply standards of excellence.

- Computers used in the classroom may endanger effective communication between students. Yet typically students tend to work in groups with only one student sitting at the keyboard while the rest of the students in the group look on. The students looking on share ideas and give advice and encouragement to the student at the keyboard, with brighter students frequently helping others.

- Computers cannot replace first-hand experience with real objects. However, computers themselves are increasingly becoming part of our everyday life. It is increasingly obvious that all students will have to be computer literate, and that we must teach students to see the computer as another helpful tool. Computers should be considered as just another medium such as books or films, with the exception that the computer interacts with the student.

The computer supplements other methods of teaching rather than replaces them. Any resource for teaching is used by a teacher in his/her own way by utilising his/her own professional skills. The computer is a flexible resource for teaching and learning, and its use should be thought about and prepared for in the same way as any other resource. Computer tutorial programs should be considered *aids* rather than replacements for human teachers.

Computer-based learning alters rather than eliminates the role of the human teacher. As with other forms of teaching, the teacher needs to find a balance between fostering self-directed learning and intervening at appropriate times. It becomes arbitrary as to who is teaching, and who is learning. Learners become teachers of their peers, and teachers become learners of the students needs.

To provide an ideal learning environment, one has to provide an instructional system which is capable of conveying the concepts to be taught, and at the same time provide an atmosphere that will motivate the student to explore and learn by discovery. For example, Sutton-Smith (1979) makes the following comments:

- it is believed that students should not be subject to direct teaching;
- it is thought that students should learn at their own pace, and learn from their own mistakes;
- although students progress at a variety of speeds it has been concluded that every student passes through the same set of milestones and in the same sequence;
- it has been discovered that a student's level of activity is a direct function of the intricacy of the object the student is involved with;
- it has been found that the activity that a student is involved in must keep the student's interest in order to encourage exploration;
- it appears that students are motivated more by their own accomplishments rather than by direct praise; and
- it is believed that a student must be concerned with the process as well as the result of the activity.

Not all students learn at the same pace. They learn in quite different ways, and have quite different capacities of perception. This means that the chance to imitate and to experiment are important. A CAI system allows the students to learn and experiment at their own pace. Mitzel (1970) listed what should be present in an instructional system offering individualised instruction. Each student should be able to:

- work at his/her own pace;
- begin and end the lesson when convenient; and
- begin at a point appropriate to his/her past achievement.

In addition to the above facilities, the instructional system should also exhibit the following characteristics (Mitzel, 1970) (Sleeman and Brown, 1982):

- provide the student's preferred mode of presentation;
- provide the student's preferred type of reinforcement;
- record deficiencies in the student's skills and knowledge; and
- react to immediate past history of responses.

The author decided to adopt both the tutoring teaching strategy and the discovery learning teaching strategy when developing the CAI package. Some of the reasons for this decision are outlined below.

Learning is not passive. The tutoring teaching strategy guides the student through a series of tasks until the student grasps the meaning of the lesson. Ideally there is no predetermined sequence, rather the student's response determines the next step. If the student understands a concept, the next segment is presented. If the student has not grasped a concept, the instruction branches to remedial material. By using the tutoring strategy the instructional system can allow for different speeds and methods of learning, and can be tailored to the needs and abilities of an individual student.

It is important to allow the students control to learn in their own way and develop their own learning styles (Noss and Tagg, 1985). Only the discovery learning strategy gives the student this level of control.

2.8. Conclusion

In this chapter, the author has carried out a survey of the contemporary teaching strategies, associated teaching tactics, educational paradigms suitable for building computational instructional systems, the use of LOGO to teach Geometry concepts, and how all these information contributed to the design of the CAI system developed in this thesis. One of the main findings of this literature survey is that to provide a effective environment for students to learn the subjects LOGO and Geometry, tutorial and discovery learning strategies are the most suitable to be implemented in the CAI system. The tutorial teaching strategy is used to teach LOGO and Geometry skills. As the student is presented with new information and tasks to solve, the CAI system will record the students solutions. The collection of this data will allow the teacher to identify common mistakes made by the students and to improve the lessons in the CAI system accordingly. Discovery learning teaching strategy will allow the students to be in control of the system and take responsibility for their own learning. Discovery learning will also provide an opportunity for collaborative learning as students will be able to work in small groups. The use of both teaching strategies will provide a rich environment for collection of student performance data for future research, especially in identifying the student's learning process, and their common misconceptions.

In Chapter 3 the author specifies the way a CAI system must be designed to maximise the benefits of using CAI, particularly in the collection of data for future research purposes.

Chapter 3

Architecture of the CAI System

3.1. Introduction

In this chapter, the author will outline a set of requirements that a CAI system for teaching LOGO and Geometry must satisfy, to enable the system to:

- successfully teach students on all levels of expertise;
- give the teacher control over what the system teaches and how it is taught; and
- be used for research purposes.

Section 3.2 will attempt to identify a detailed set of requirements of a CAI system. Based on these requirements, the author will propose a conceptual model of a CAI system in Section 3.3. This model is then evaluated to ensure that it satisfies all the necessary requirements in Section 3.4. Finally, Section 3.5 concludes this chapter.

3.2. Requirements

In this section, the author will outline a series of requirements that must be satisfied by the CAI system so that it can be effectively utilised by students, teachers, and researchers. The requirements are divided into three sub-categories:

- requirements and functionalities to support students and foster learning;
- requirements and functionalities to enable the teacher to set up lessons; and
- requirements and functionalities to enable the researcher to set up experiments and to collect data.

To support students on all levels of expertise a CAI system must possess the following functionalities:

- provide an enjoyable and motivational experience for the students involved;
- allow students to learn at their own pace and learn from their own mistakes;
- take the initiative when a student's weaknesses become apparent;
- ensure the basic concepts are properly understood by the student;
- accept a range of solutions to a problem;
- be able to explain to each student exactly where he/she has gone wrong;
- provide on-line help;
- teach the students the best way to solve a particular problem;
- be able to monitor an individual student's progress and adapt the instructional process accordingly (i.e., if the student does not grasp a concept, then the system may want to present remedial materials);
- provide students with the opportunity to test and share ideas;
- be able to record the end point of each session, thus enabling the student to continue from their last point of stop;
- have clear explanations for the common misconceptions;
- allow the student to build a library of his/her own programs; and
- attain the right mixture of an exploratory learning style through learning by doing (Papert, 1980) and a guided learning style through teaching assistance (Sleeman and Brown, 1982).

Further to the above requirements, to allow the teacher to have control over the subject content and method of teaching, a CAI system must exhibit the following functionalities:

- be able to be used in a variety of subject areas, and be as general purpose as possible;
- give the teacher the opportunity of incorporating his/her teaching tactics into the material;
- allow lessons to be modified quickly and easily;
- be able to maintain a history of the students problem solving processes, thus enabling the teacher to analyse these histories and determine the misconceptions of the students;
- have a facility to accumulate explanations for common misconceptions; and
- be able to teach with examples and explanations.

To be suitable for research purposes a CAI system must demonstrate the following functionalities:

- be able to be set up to conduct experiments such as testing different teaching methodologies, and methods of remediation, thereby evaluating the instructional component;

- be capable of capturing data such as common errors, time taken on tasks, and number of students successfully completing a task (particularly on the first attempt);
- provide access to data to help the researcher develop a model of each student's performance;
- highlight areas where learning difficulties are encountered; and
- be able to measure how successful a lesson is in communicating with the student.

3.3. Conceptual Design

There are two phases of learning in the CAI system. The first phase involves the teaching of LOGO. Here the CAI system will attempt to teach the functionalities of the graphical programming language called LOGO. To achieve this, the CAI must contain a set of lessons for teaching LOGO, appropriate remedial lessons, together with explanations on the common misconceptions of concepts within these lessons. The second phase will involve the teaching of the subject area Geometry. For this phase as well, there should exist a sequence of lessons, appropriate remedial lessons, and a set of explanations for the common misconceptions within this subject area.

To satisfy the system requirements listed in Section 3.2 the conceptual design outlined in Figure 3.1 has been adopted.

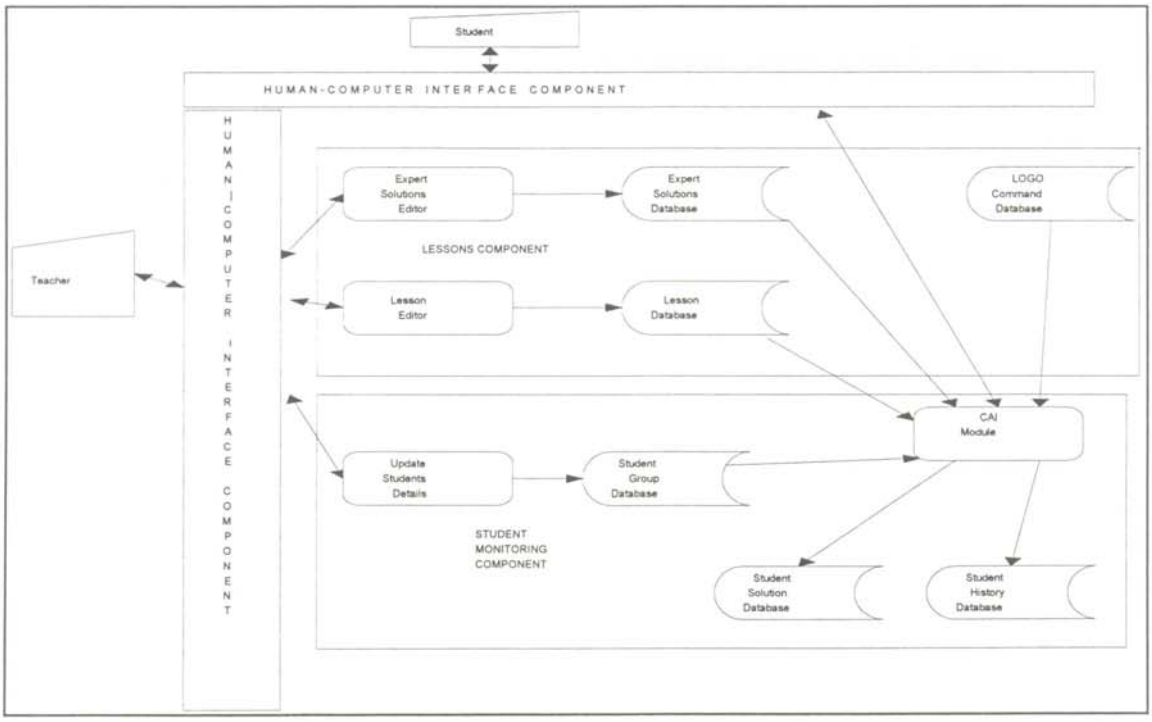

Figure 3.1 : Conceptual design of the CAI system

The CAI system has three major components. They are:

- Lessons Component;
- Student Monitoring Component; and
- Human-Computer Interface Component.

The following subsections will describe each of these components.

3.3.1. Lessons Component

The Lessons Component consists of the information the system proposes to teach. It is constructed from the knowledge of experts (teachers), and is comprised of three different types of knowledge:

- Domain Knowledge: As previously mentioned, in Phase I, the students must learn the computer language LOGO, and from this stage they may progress onto learning other subjects. If errors are made in the LOGO programming while working in another subject area it is important that they are corrected. Consequently the system includes a database containing the commands and semantics of the LOGO programming language. The information in this database remains static. This database is called the "LOGO Command Database".

- Curriculum Knowledge: This knowledge base consists of a lattice of lessons connected by prerequisite relationships. Each lesson consists of an example, a description, a task that must be completed to see if the topic was understood, and a list of common misconceptions so that explanations may be generated as to why a student's solution did not match one of the expert's solutions. The list of common misconceptions are stored in a database, and this database is built by the teacher over a period of time Curriculum knowledge is stored in the "Lessons Database".

- Expert Knowledge: The system allows "expert" solutions for each task to be stored. The expert enters the "best" solution to a problem first, and this is displayed as the correct answer. This satisfies the condition that the system must be able to demonstrate the solution to the task being taught, and furthermore should solve a problem using the method that it would like to promote as the best method. The set of solutions to a task are organised in terms of the choices available at each state of the problem. The system is also capable of determining when to proceed to a new topic with the student. It contains procedures that can judge whether a student's answer is correct or not. That is, whether a student's solution truly matches one of the expert's solutions. Consequently, the system can trace through the student's problem solving actions, and communicate the appropriate solution. Expert knowledge is stored in the "Expert Solutions Database".

When designing the curriculum knowledge the teacher enters a series of "lessons" by means of a Lesson Editor. The original lesson and all remedial lessons for a topic are placed in one group, with remedial lessons often aimed at a different level of difficulty from the original lesson. If a student successfully completes a lesson in a group, the system automatically displays the first lesson in the next group. Not only can the lessons themselves be altered but the order in which they appear can be changed, and lessons can be inserted and deleted at any point in the sequence. This arrangement permits the teacher to easily make improvements and modifications. This helps them to meet the guidelines for determining the content and sequencing of instruction set by Gagne (1985), where target knowledge is broken into increasingly smaller units of knowledge without duplication and any harmful interactions.

The CAI system allows for learning by viewing worked examples and through explanations of the theory. Overall, it is the teacher who determines the content and sequence of the instruction.

The Expert Solution Editor allows a set of "expert" solutions to be entered for each task. The expert should enter the "best" method of solving the problem first, as this will be displayed as the correct answer if the student fails to find a solution or near solution to the problem. The information is organised in terms of the choices available at each stage of the problem, as shown in Figure 3.2.

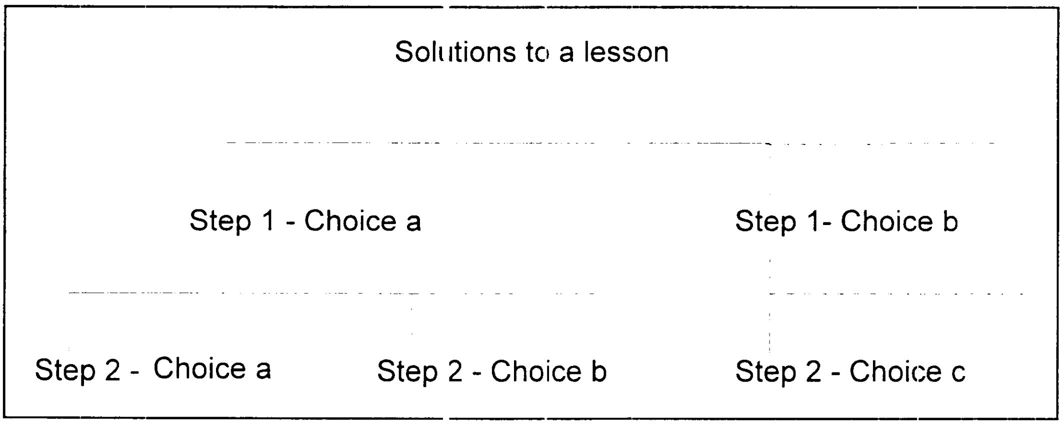

Figure 3.2 : Organisational structure of the solutions to a lesson

3.3.2. Student Monitoring Component

The Student Monitoring Component records the student's developing proficiency, the student's common errors and misconceptions, and the student's abilities and preferences in terms of a particular learning style. The student's progress is represented in minute detail and is updated dynamically during the course of the tutoring discourse. This information is stored in the "Student History Database".

Research in the area of CAI systems has many aims. Two of the aims which are addressed in this thesis are:

- to improve incremental learning processes using knowledge-based systems, and to acquire a better comprehension of how people understand, learn, and operate complex systems; and
- to evaluate empirically the effectiveness of the student support systems built and to determine possibilities for improvement.

To achieve the above aims, the following issues need to be investigated:

- What are the general principles that determine the correct combination of free exploration and instruction?
- How can we ensure that systems take the initiative when necessary but are non intrusive?
- How can we make systems flexible to the requirements of individual students?
- How can explanations be adapted to fit the student's conception of the task?
- What is the role and relative importance of verbal and nonverbal (e.g., pictorial) explanatory material, and when should one be chosen over the other?

The Student Monitoring Component is used to help researchers construct a model of each student's skills and investigate the above issues. The student model helps in understanding the student's process of reasoning as the student progresses through the CAI system, and the data collected can also be used to evaluate the Lesson Component of the CAI system.

The Student Monitoring Component is used to decide what the system should teach in different circumstances. It allows the instructional interaction to be adjusted for the individual student. The CAI system uses the knowledge about the student's rate of success and the lesson goal structure to decide which instructional activities will be presented next. The Student Monitoring Component records details of what material has been covered and what teaching tactics have been used. Information stored in the Student History Database is used as feedback to manually revise the Lessons Component.

When students are left to their own devices to solve a problem, they are likely to employ their own favoured problem solving style. Such conditions allow different styles of working to be observed and researched. Some students will rely on pictorial representations to recognise the problem, whereas others will make use of linguistic identifications and descriptions. In addition, the fear of being wrong can sometimes lead to a change of goals in order to avoid mistakes.

Not only does the CAI system record details about each student as they use the system, but students can be placed in various categories, enabling comparisons to be made between them by allotting the students into a variety of groups. A student may belong to one or more groups, and each group may contain any number of students. A researcher may group females and males, or group together students with a range of learning abilities. The grouping of the students which may or may not be random, records the members of an experimental group. This facilitates using the CAI system to monitor students and acquire data relevant to specific research questions.

Each individual student can be given a specific lesson on which to start, and a record is always kept of his/her "next" lesson in the sequence. The CAI system also records the number of attempts a student makes at each lesson. The next lesson in the sequence is determined by the system in order that a student may repeat or skip lessons based on their performance. This is of most use when remediation is required.

3.3.3. Human-Computer Interface Component

Nwana (1990) noted that there is another important component in a CAI system, namely the Human-Computer Interface Component or communication component, which controls the interactions between the system and the user (who may be the student or the teacher). The Human-Computer Interface component is concerned with the way the CAI system is presented to the user, and the way the user must respond.

3.4. CAI Knowledge Base

In summary, the knowledge stored in a CAI system ought to consist of the following:

- knowledge of the subject to be taught. This knowledge restricts the number of possible actions, and describes rational goals and operations;
- knowledge of the student, and the student's performance. There are many types of students, and the needs of an individual student grow with experience;
- knowledge of suitable teaching skills. The system must incorporate instructional strategies that are based on pedagogical theories; and
- theory of how to apply teaching skills in particular cases. The system must be aware of the student's most common problems, and the information structures that manage the communication processes should be made explicit, so the student controls the system.

The above knowledge is stored by means of several databases. Entity-Relationship (E-R) diagrams are used to depict the conceptual schema of each of the databases (Date, 1995). Details of all the database entities can be found in Appendix D. The designs of all the databases are shown below:

- LOGO Command Database: This database contains the sets of valid LOGO commands, arithmetic operators and editing commands. The E-R diagram of this database is shown in Figure 3.3.

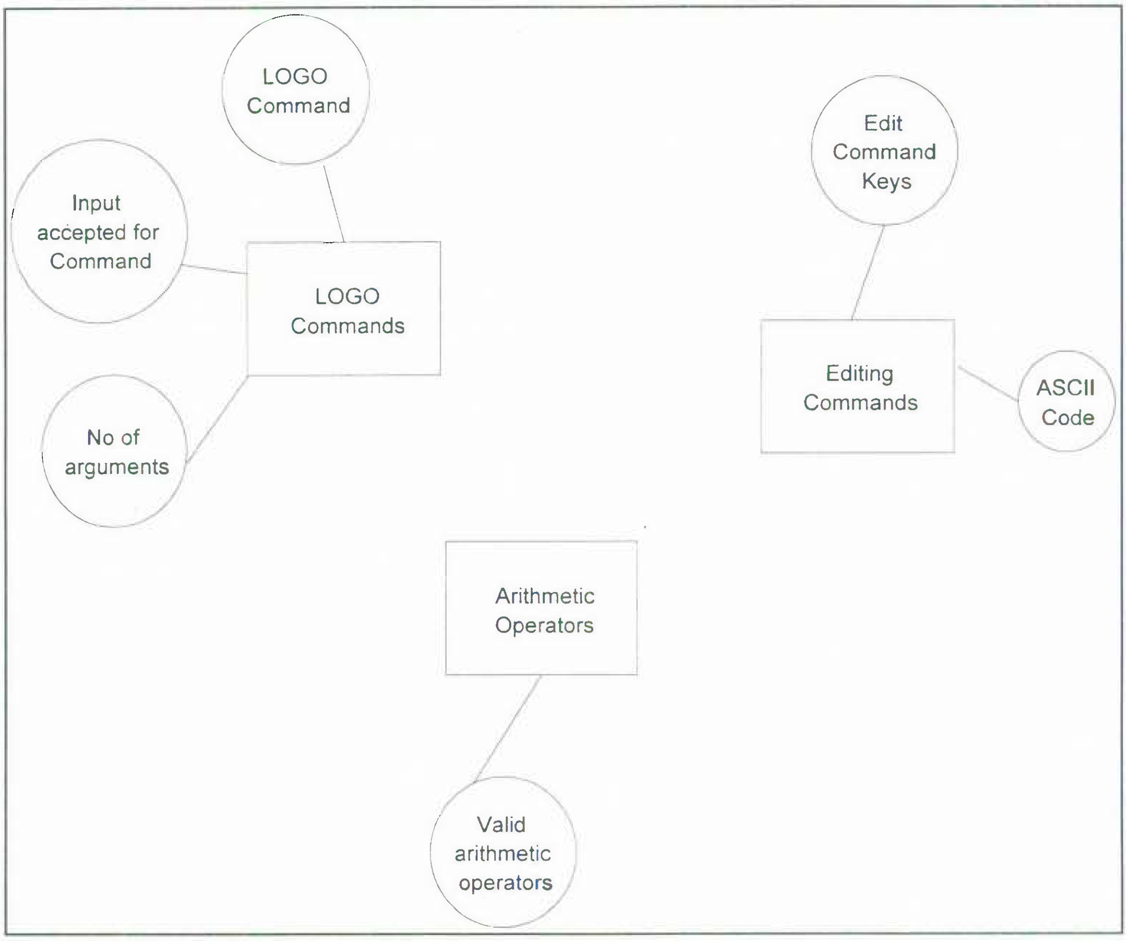

Figure 3.3 : E-R Diagram for the LOGO Command Database

- Expert Solution Database: Each of the problems set must have a set of expert solutions. Where the expert solution contains variables the Expert Solution Database contains default values for these variables. When the solution relates to a Geometrical shape the database stores information in regard to the shape's sides and angles. The E-R Diagram of the Expert Solution Database is shown in Figure 3.4.

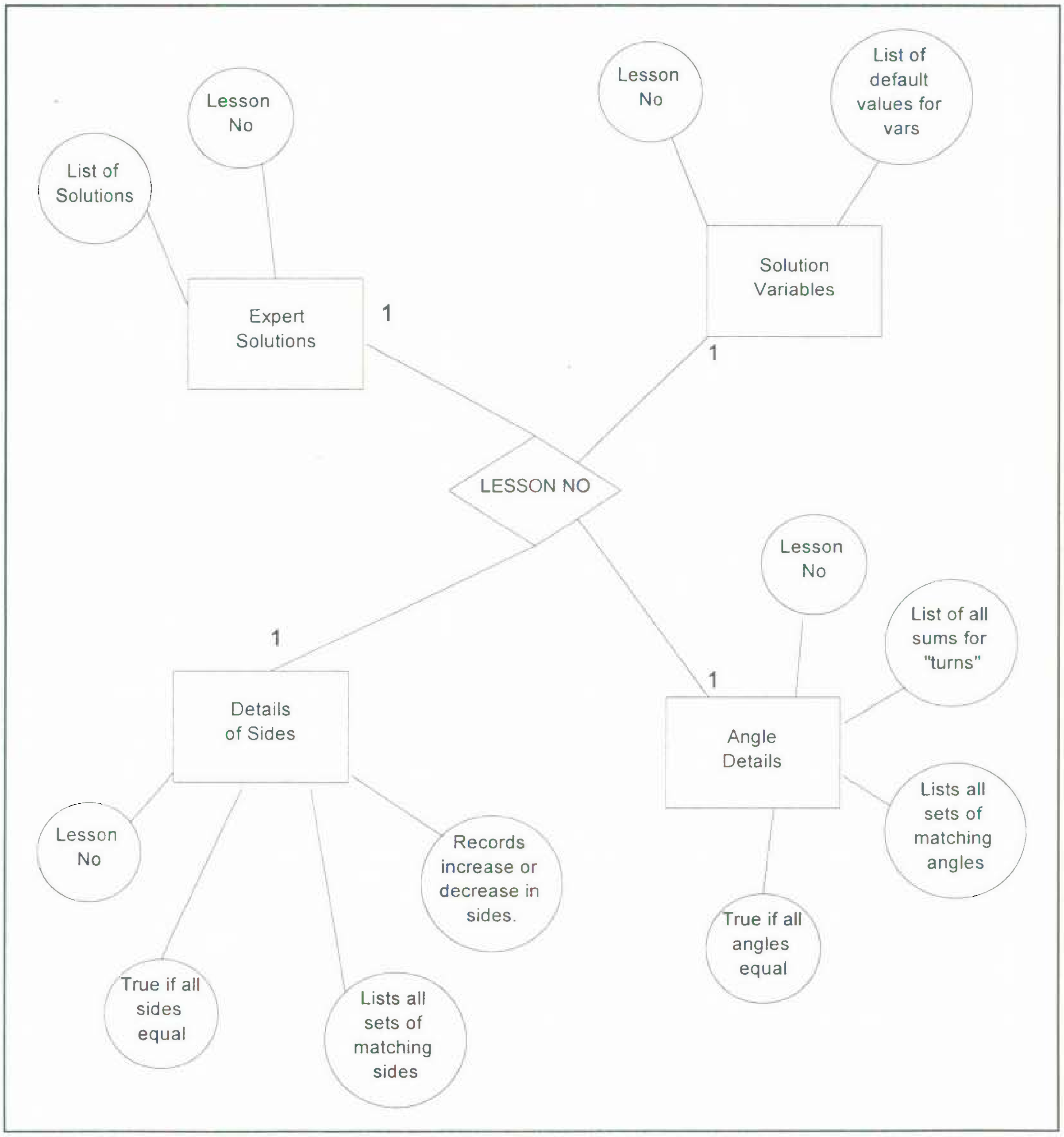

Figure 3.4 : E-R Diagram for Expert Solution Database

- Lessons Database: This database maintains lessons, remedial lessons and common misconceptions for different subject areas. Figure 3.5 shows the E-R diagram for the Lessons Database.

Figure 3.5 : E-R Diagram for Lessons Database

- Student Group Database: This database contains the student details in their experimental groupings. The E-R Diagram is shown in Figure 3.6.

Figure 3.6 : E-R Diagram for Student Group Database

- Student History Database: This database contains the problem solving sequence indexed by individual students. The E-R diagram is shown in Figure 3.7.

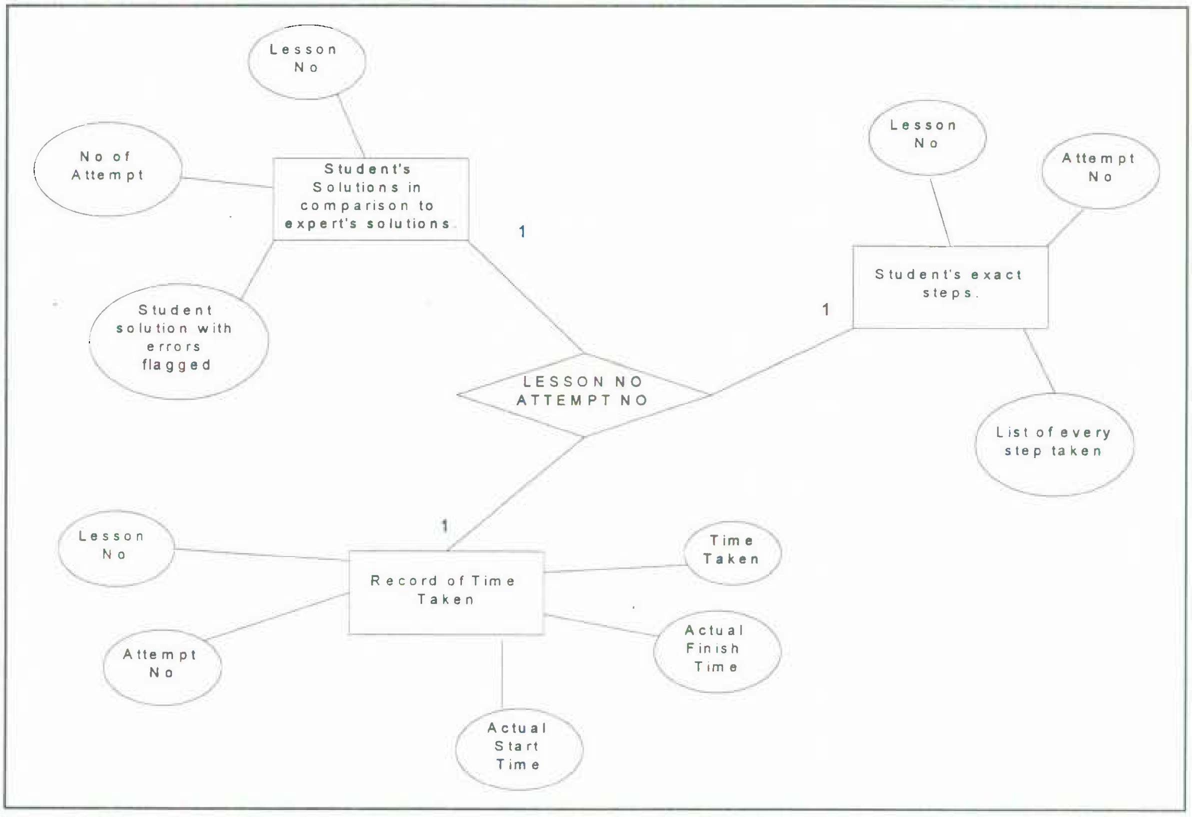

Figure 3.7 : E-R Diagram for Student History Database

- Student Solution Database: This database consists of a library of programs maintained by the students. The E-R diagram is shown in Figure 3.8.

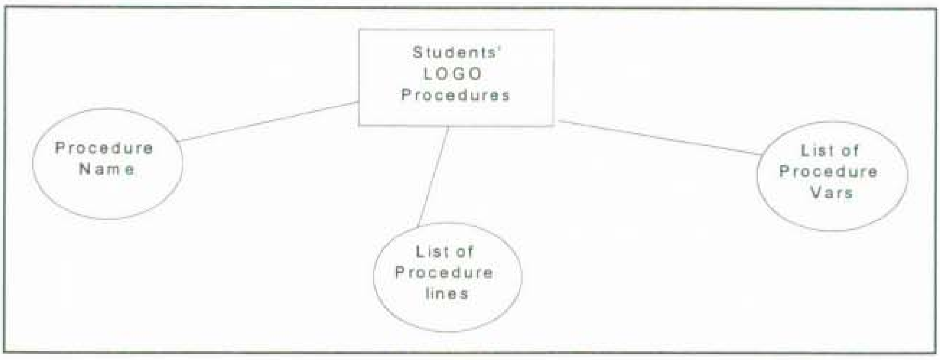

Figure 3.8 : E-R Diagram for Student Solution Database

3.5. Design Evaluation

The CAI system provides a motivational learning environment which guides the students through the process of correcting their own misconceptions by advising them when they make mistakes. The system makes it easier for the student to grasp basic concepts by not presenting too much information at once. It will accept alternative solutions to a problem, but it always aims to promote the expert method of solving the problem.

The CAI system is designed to give the student control by leaving the student to learn at his/her own pace, by giving the student the choice of whether or not to proceed with a new lesson, and by giving him/her the option of acquiring additional information through on-line help. However the CAI system is also capable of giving advice to the student or presenting remedial material. It monitors the student's activities and any current problems or misconceptions, and records the exact place where the student experienced difficulties. Thus, the system is able to adapt to an individual's learning needs by presenting remediation in terms of what it believes the student does and does not know. The system can be made to adjust the sequence of the lessons to meet changes in a particular student's performance.

According to Anderson, Boyle and Reiser (1985), a system should stop students from going too far from the optimum solution path. On the other hand, they also concede that, unless students flounder they will never learn metacognitive strategies. Further, they conclude that a mixed approach is probably best, to intervene when the student is learning, but still give the student the opportunity for free exploration. The time students spend in free exploration allows them to test their own ideas, and gives the teacher the opportunity of observing their preferred working styles. The CAI system developed aims to find a balance between intervention and exploration by providing both modes, and allowing the student to choose in which mode he/she wishes to work. In either mode the students may save their solutions for later use.

By examining the data collected in the Student History Database during a student's early experiences with using the CAI system, the teacher can develop a new set of lessons which better help the student cope with new problems by:

- making suggestions suited to the student's precise requirements; and
- tailoring the content of the remedial lessons to meet the particular student's needs.

The teacher is able to change the subject area (as long as the new subject area can be taught using LOGO), and the teacher is able to design the content and sequence of the lessons in the subject area. Both examples and explanations can be used in constructing the lessons, and using the lesson editor the lessons can be changed quickly and easily.

Sometimes it is impossible for the teacher to predict every correct answer from the students, and this results in earlier versions of the CAI system considering some answers to be wrong. Similarly, students can sometimes give wrong answers not envisaged by the teacher. If many students give the same incorrect answer the system should give specific feedback for the error. By examining the contents of the Student History Database, the teacher can identify and amend such problems, thus improving the Lesson Component for future use.

The CAI system can not only be used to discover the gaps in a particular student's knowledge (and therefore make the teacher aware of where remediation is required), but over a period of time it presents the opportunity for the teacher to acquire a better comprehension of how the students learn a particular topic, and to establish if there are any patterns in the students learning behaviour such as:

- do males grasp LOGO/Geometrical concepts better than females?
- do CAI systems appreciatively favour particular types of learners?
- what differences can be found in students learning to use the system individually or in groups? If groups are used what is the best size and composition for a group?

3.6. Conclusion

In summary, an ideal CAI system should know what to teach and to whom, as well as when and how to teach it. The teacher must be able to tailor the content and method of instruction to suit a student's individual learning pattern, and the student must not be limited to a set of predetermined actions. A CAI system must understand the nature of a student's misconception and present remediation accordingly, rather than simply repeating the set of instructions covering the content not mastered. Finally, the system should be able to collect the relevant data so that the teacher can discover the best learning strategies to use in the subjects being taught.

The author has explained the ways in which the designed CAI system attempts to meet the above requirements. In the following chapters the author examines the various components of the system in detail (i.e., the Lesson Component, the Student Monitoring Component, and the Human-Computer Interface Component).