Int. J. Engng Ed. Vol. 10, No. 3, pp. 299–305, 1994
Printed in Great Britain.

0742–0269/91 $3.00+0.00

Explaining and Understanding Engineering Problems—An Intelligent Tutoring Approach*

BENITA COX
Imperial College of Science, Technology & Medicine, London, UK

Recent years have seen an increased awareness of the potential of intelligent tutoring systems (ITS) as an innovative driving force in engineering education. In particular, these systems provide an opportunity not only to improve the user's problem-solving ability but also to increase his or her understanding of a given problem domain. This focus on increasing the user's understanding of a problem domain is the basis of the explanation-driven understanding-directed (EDUD) framework for ITS design described in this paper. The basic premise behind this framework is that effective solving of engineering problems is facilitated where there is understanding of the relevant first principles. Engineers whose training is too closely related to current solutions of technological problems with insufficient knowledge of first principles lack both flexibility of mind and the background knowledge to contribute effectively to the solution of new problems. The EDUD framework is designed to support the user at three broad levels in the learning process, namely, knowledge of basic principles, acquisition of problem-solving skills and expertise. The user is represented at each level in a user model based on cognitive principles which reflects his understanding of the level. For each of the levels the target knowledge is segmented and structured so that assimilable 'chunks' of pertinent information are presented to the user. An example of how the framework is used in structuring and presenting knowledge about a car's fuel system is described.

INTRODUCTION

ARTIFICIAL intelligence (AI) provides a means of solving ill-structured problems. Many engineering problems are of such a kind and the interest of the engineering community in AI has been considerable, particularly in the area of knowledge-based expert systems. AI can, however, also make significant contributions to the teaching of engineering knowledge. The nature of most engineering problems requires the engineer to have sufficient understanding of the domain to carry out careful cross-checking and testing of solutions. This means that problem-solving decisions frequently need to be traced to first principles to make sure that they are soundly and demonstrably based in fact. Thus the factual knowledge relevant to a particular engineering problem forms a foundation upon which problem-solving skills can be built [1]. Understanding such knowledge is essential to successful problem-solving. Yet, the approach commonly used in existing intelligent tutoring system (ITS) design emphasizes the teaching of problem-solving skills rather than that of first principles. Furthermore these problem-solving skills are viewed as specific to a particular task [2]. Present ITS design examines a task and infers the underlying cognitive processes necessary for the problem's solution. Specific problem-solving skills are thus firmly linked to a specific task and it is these skills which are tutored with the help of the system. However, if the user is to recall and apply the knowledge acquired from using the system, then it should be designed with greater emphasis on ensuring understanding of the basic principles rather than exclusively on improving particular problem-solving capabilities. This requires the ITS to adapt its performance to the user's developmental level at a given cognitive stage. In order to achieve this a profile of the user's current understanding levels needs to be constructed which reflects the user's initial knowledge of the domain as well as his/her progress in a learning task brought about by his/her interaction with the system.

The explanation-driven understanding-directed (EDUD) framework represents the user's level of understanding of a given body of domain knowledge and links the explanations given with this representation. Structuring of the target knowledge is based on the categories of understanding which provide units for explanation [3–5]. Each stage in the process of understanding is thus linked to a previously defined explanation category, providing a set of structures, amenable to both computation and human comprehension. The sequencing of the presentation of these categories of explanation is arranged so as to increase the user's understanding of the domain. The ITS based on this framework is capable of functioning in three distinct modes: a

* Paper accepted 27 April 1994.

directed mode which teaches basic principles; an interactive mode which facilitates problem-solving; and an expository mode which demonstrates expert reasoning.

THE IMPORTANCE OF THE CONCEPTS OF UNDERSTANDING AND EXPLANATION IN ITS DESIGN

It has previously been stated that the concepts of understanding and explanation are seen as fundamental to the design of instructional engineering systems. Yet, frequently these terms are used without sufficient attention being given to their meaning. A brief discussion therefore follows as to what is meant by these concepts and how they may be addressed in the design of the teaching system.

The concept of understanding

In considering the concept of 'understanding' in relation to instructional design, a distinction must be drawn between the notion of understanding as a state, i.e. as a practical manifestation demonstrated in particular subject or task areas, and that of understanding as a general process, i.e. of moving from one knowledge state to a higher knowledge state.

Educational theory provides guidelines as to how the state of understanding can be measured, which has implications for pedagogic design. An example of this approach is found in Bloom's work which provides a taxonomy consisting of categories for six cognitive abilities, namely knowledge, comprehension, application, analysis, synthesis and evaluation [6]. Understanding is linked to these categories in such a way that the attainment of the educational objectives described in each category is indicative of a particular level of understanding. For example, 'some understanding' of a situation or phenomenon is demonstrated by the ability to 'translate' it, i.e. by describing the initial phenomenon in terms slightly different from those originally used. Deeper understanding is reflected by the ability to interpret the information, i.e. to summarize and explain the phenomenon previously described. A particularly interesting feature of such a taxonomy is that is not task-specific but domain independent and may therefore be used as a guideline for achieving educational objectives in a variety of tasks and domains. As such it is conceptually amenable to be included as a measurement of a user's level of understanding in software design.

However, in designing instructional software the focus is not only on providing measures of the user's current state of understanding, but with facilitating the process of understanding. This is commonly addressed in computerized engineering teaching through the provision of simulations which provide an environment where the process of understanding is facilitated through active involvement in exploration and discovery learning, a technique well recognized in educational theory. However, it is also recognized that this technique is not without its drawbacks. Learning through exploration places high cognitive demands on the learner [7]. Without sufficient assistance the learner may flounder and miss the richness of information provided by the simulation. Instructional software must ensure the transfer of knowledge in such a way as to achieve epistemic continuity for the user [8–10]. Simulations, therefore, whilst having a valuable role to play in improving problem-solving skills, frequently ignore the teaching of underlying first principles and the user's current level of understanding of the target knowledge.

The concept of explanation

Much research in the field of 'explanation' has been undertaken in the design of expert systems. However, lessons learnt from this field must be viewed with care when applying their results to instructional design [11]. Firstly, there is a fundamental difference in the purpose for which explanation facilities are provided in the design of expert systems and that of teaching systems. In an expert system the goal of the explanation is to explicitly provide justification that the knowledge used in the reasoning process as well as the process itself is correct and appropriate. It is therefore concerned with communicating to the user what the system itself 'knows' in terms of how and why a particular conclusion is reached on a particular problem-solving task. Because the emphasis is on justifying conclusions arrived at by the system at expert level, the issues with which research in this area are concerned include how an expert decomposes and integrates his/her reasoning in a problem-solving context and how this may be articulated in system design.

The goal of the explanation in instructional software is different to that described above. Here the aim is to increase the (non-expert) user's understanding of a particular domain, and to do so in such a way as to ensure that the user may recall and apply that knowledge at a later date. The explanation process is therefore part of an act of teaching which must be related to the conclusions that the user expects the system to draw and the way in which he/she would have drawn those conclusions him or herself. The basic emphasis is not, therefore, on how and why a particular conclusion was reached by the system but on what, when and how to explain the given body of target-knowledge. Thus the primary concern of explanation is to enhance the user's learning/understanding of the target-knowledge, a task which requires information of the user's current epistemological state.

This requirement has important implications for the design of the architecture and behaviour of instructional systems. It cannot be assumed that explanations will be given in response to some content-seeking question initiated by the user and that the user knows the correct question to ask. The explanation facilities in expert systems may be described as playing a passive role. In an instructional system on the other hand, explanations must play an active role since it cannot be assumed that the user will request an explanation at the appropriate time or level, nor that the explanation-seeking question asked will be the correct one. The software must be able to determine the appropriate content and timing of the explanation which necessitates the explanation facilities being extended to select appropriate material, adapt this material to the user's level of functioning and to conduct an efficient interaction. In those instances where a question is initiated by the user, the system is required to consider, inter alia, what inferences the user is trying to draw and what his/her problem-solving procedure is to accomplish his/her task. The initiative for providing explanations is therefore shared between the system and the user.

THE EDUD FRAMEWORK

The EDUD framework consists of the following modules:

- a knowledge base containing the structured domain knowledge;
- a user model, which represents the user's level of understanding; and
- a dialogue control module which regulates the interaction between the user and the system.

The EDUD knowledge base

Central to the EDUD approach is the condition that knowledge about a given problem-domain can be structured according to some well-grounded principle. The object is to organize the domain knowledge into several topic-dependent, autonomous information-containing structures, each of which can act as a context for the explanation-seeking and explanation-giving process. Engineering knowledge is particular amenable to structuring. Much work has already been done in representing engineering knowledge in structured formats. The EDUD framework builds on this approach. Knowledge is structured into a hierarchical framework consisting of five major categories. Each of these categories is identifiable with a particular type of explanation. These are:

1. identity explanations, which are concerned with the definition and classification of objects;
2. functional explanations, which are concerned with the intended functions of objects in the domain;
3. causal explanations which are concerned with cause/effect links;
4. complex-derivational explanations which are concerned with defining how a domain relates to all other domains with which it is functionally linked; and
5. hypothetico-deductive explanations which consider relationships both actual and potential.

The representational formalism used to express the domain knowledge in the EDUD approach is a frame-based system which is a complete semantic network. The general structure of a network for the fuelling system of a motor car is shown in Fig. 1. The hierarchical structure of the semantic network as a whole enables limited inferences to be drawn, e.g. that the throttle-valve is part of the fuelling system, which in turn is part of the engine system. The conclusion can therefore be reached that the throttle valve is part of the engine system or that the float chamber is part of the carburettor.

The EDUD user model

To be an effective tutor the system needs to adapt its performance to the user's developmental level at a given cognitive stage. In order to achieve this a profile of the user's current understanding/learning level needs to be constructed. This profile reflects the user's initial knowledge of the domain fragment of an application domain intended for teaching as well as his/her progress in a learning task brought about by his/her interaction with the system.

In the EDUD framework the user's understanding of the material presented to him/her is assessed at each level in the hierarchy. Based on this assessment a decision is made as to what material should be explained to him/her next. The goal is to pinpoint the conceptual relations which are understood at a specific phase of understanding and consequently 'predict' those relations which still need to be tutored and those which must have already been mastered in order to achieve the present level of understanding.

This approach has important implications for system design. It allows direct comparison of the user's input with the steps in the hierarchical structure. In this way, detailed inference models of the processes taking place within the user are obviated, as are any cumbersome computational overheads that might be associated with them.

The stages of understanding through which the individual passes reflect and are linked to the categories of explanation. These are:

- The Stage of Figurative Knowing.
- The Functional Stage.
- The Cause/Effect Stage.
- The Complex Derivational Stage.
- The Stage of Hypothetico-Deductive Reasoning.

The Stage of Figurative Knowing During this stage of understanding the learner grasps the definitional attributes of a particular object or concept. Where explanation is required the goal of the explanation is to ensure that the user understands the basic definitional characteristics and attributes of a concept in a specific domain together with their value. The explanation process is commonly initiated in this stage by questions such as 'what is . . . ?', 'is it . . . ?', 'how much is . . . ?', and understanding-directed by 'it is . . .'

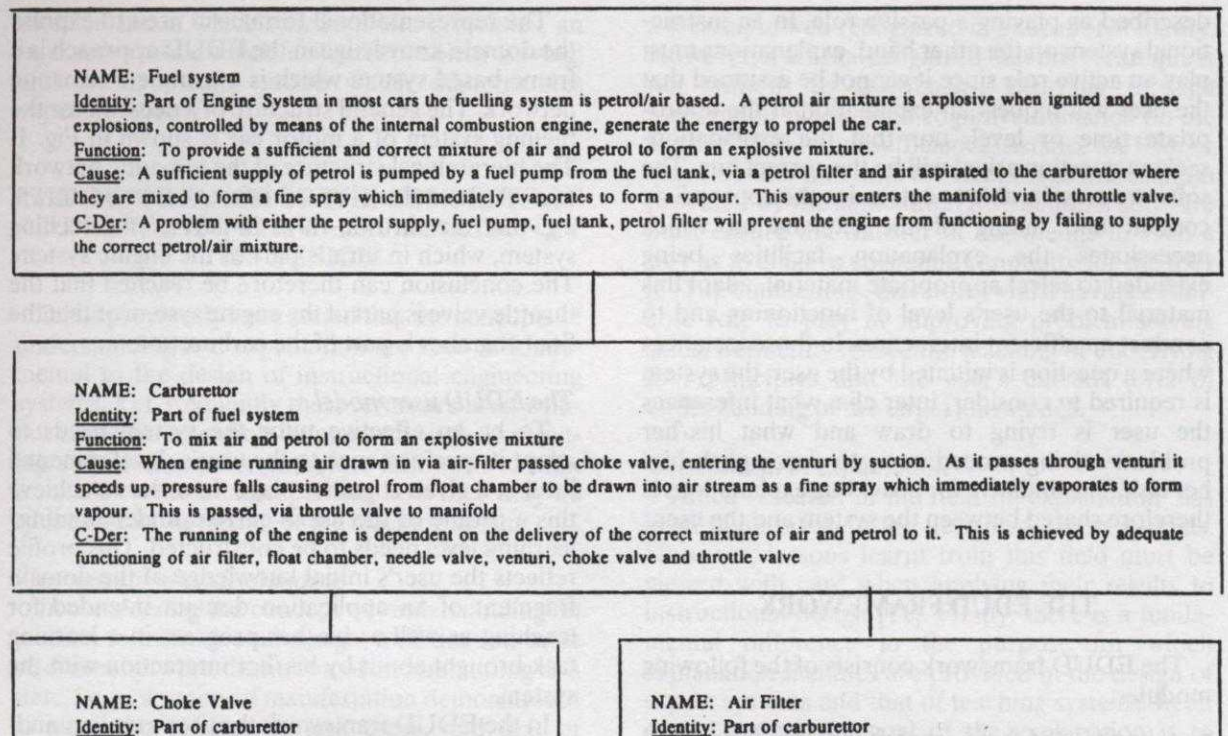

Fig. 1. A section from the frame-based representation of knowledge about a car's fuel system as used in the EDUD framework.

The Functional Stage Understanding at this stage requires a knowledge of the purpose or use of an object or action. Attention should be paid at this stage to the difference between 'cause' and 'function' which are not synonymous. Functionality differs from causality in its attention to goals. In many instances a functional relationship underlies a causal chain which constitutes a procedure. The explanation question here is 'for what purpose . . . ?' and the understanding directed by 'for the purpose of . . .'

The Cause/Effect Stage At this stage the pertinent cause/effect relationships between entities are understood as well as distinctions between direct causal relationships, i.e. where there is no ambiguity about the cause/effect link and relative causal relationships, where the entity represents an object which may have causal influence under particular conditions. Explanation may be sought at this level by asking 'how . . . ?' and understanding directed by 'by . . .'

The Complex Derivational Stage Understanding at this level requires a grasp of why it is the case that a particular set of conditions exist. Explanation is sought by attempts to discover 'why it is the case that . . . ' The goal of the explanation at this stage is to ensure that given an effect the user may infer a cause. This is possible only if all the prior definitional, functional and causal relationships have been understood.

The Stage of Hypothetico-Deductive Reasoning In this stage the results of the prior phases are used as propositions which are then further operated on. That is to say, various kinds of logical connections are made between them (identification, disjunction, conjunction, implication, etc.). In this stage the user applies his/her previously acquired information directly in a new situation. Various pieces of information are compared and explanation is often induced by the question 'what if . . . ?' and understanding-directed by 'if . . . then'.

The final stage is that which comprises expert reasoning. Here the expert uses both his/her theoretical knowledge of the domain together with his/her experiential knowledge to form heuristic solutions to problems.

Dialogue control module

The EDUD framework is designed to function in three distinct teaching modes. It is necessary to ensure that the user first has a level of understanding of the basic principles of the domain. Thus in the primary mode the system functions to ensure the user's understanding at the level of figurative knowing, the functional stage and the cause/effect stage. An effective method of providing 'meaning' to a new concept is to encourage the user to classify this new information in the light of what he/she already knows, e.g. by providing him/her with examples or analogous situations. The second mode addresses the complex-derivational and hypothetico-deductive stages of understanding and provides for conversion of the previously obtained declarative knowledge. This may take, for example, the form of simulations. Finally, the system must provide for obtaining expertise in a subject area. One method of acquiring expertise is through observation and interaction. The EDUD framework therefore provides for an articulate expert system.

MONITORING OF THE USER

The learning/teaching strategy is designed to reflect the chosen knowledge-decomposition hierarchy. Essential in this strategy are those concepts in the particular domain which are considered to be the most important for the user to understand, together with their hierarchical relationships with other concepts in the domain and a set of rules monitoring and controlling the user's progress through the stages of understanding.

A frame-based knowledge representation is used as a basis for modelling the user's level of understanding and determining which explanation is needed. The advantage of using frames is that each of the cognitive structures which the user must acquire in the process of understanding a problem solving space can be represented as a slot containing a knowledge 'structure'. Each frame consists of a number of slots, each slot aiming at a specific level of understanding in the EDUD hierarchy. Each slot in the frame has a value attached to it.

A slot may also contain a procedural attachment. For example, in order to obtain the value 'yes' or 'no' for the identity slot in the above frame, the procedural attachment defines a question/answer procedure. The system asks the user the question 'Do you understand what the fuel system is?' and the value in the understanding-fuel-system frame is instantiated according to the user's response (Fig. 2).

Diagnostic rules for assessment of user's understanding

Teaching of basic principles The system takes the initiative in the initial stages The interaction commences with the dialogue control module, which initiates a diagnostic test to assess the user's present level of understanding in relation to a specific concept. The user's reply is assessed and the user model is updated accordingly. When at a particular level the user states that he/she does not understand a question, no further diagnostic questions are asked. It is assumed that he/she cannot have attained a higher level of functioning in the hierarchy and all subsequent levels of explanation are offered to him/her. This is expressed in a system of general rules (Fig. 3). At any point in the explanation process where a new concept is encountered, an explanation of that concept together with its hierarchical structure is offered.

Teaching competence in problem-solving tasks In the problem-solving stage the user is set a number of problems to solve. During this phase he/she may interrogate the system. The explanations given in response to his/her queries are tailored to the system's perceived level of the user's understanding.

The explanation offered is determined by a set of general rules. For example:

Rule E1
If query is 'what is x?'
then offer identity explanation of x

Rule E2
If query is 'what does x do?'
then offer functional explanation of x

Rule E3
If query is 'how does x achieve . . . ?'
then offer causal explanation of x

Rule E4
If query is 'why is it the case that x . . . ?'
then offer complex derivational explanation of x

Facilitating acquisition of expertise A detailed account of this system is not given here other than to point out that the same principles used in the primary mode are applied here. For example, should a user ask a question, the system first establishes his/her level of understanding as a frame of reference and explanation is directed at this level.

Frame	Slot	Value
Understanding-fuel-system	identity	yes/no
	function	yes/no

Frame	Slot	Procedure
Understanding-fuel-system	identity	Question

Fig. 2. Frames for representing user's level of functioning.

Rule A1
Assess understanding of identity
If identity understood then assess understanding of functional relationships else explain identity, functional, cause/effect and complex-derivational and set problem solving tasks

Rule A2
Assess understanding of functional relationships
If functional relationships understood assess understanding of cause/effect relationships else explain functional, cause/effect complex-derivational and set problem solving tasks

Rule A3
Assess understanding of cause/effect relationships
If cause/effect relationships understood, assess understanding of complex-derivational else explain cause/effect, complex derivational and set problem solving tasks

etc...

Fig. 3. Rules for assessing understanding of basic principles.

CONCLUSION

The EDUD framework provides a practical approach to the analysis and presentation of knowledge for computerized engineering teaching. The importance of the concepts of explanation and understanding in knowledge communication and acquisition have been stressed. A distinction has been made between problem-solving knowledge and the declarative, factual knowledge relevant to a given problem domain. The latter kind of knowledge is the foundation upon which problem-solving skills are built.

The EDUD approach offers a practical, psychologically plausible and computationally implementable way in which such knowledge can be communicated and assessed to assure that it is understood. The basic organization of knowledge in the EDUD framework makes it particularly suitable for teaching engineering problems. By stressing the importance and priority of the declarative (factual) knowledge we depart from the generally adopted methodology of existing tutoring systems which are, by and large, oriented towards teaching problem-solving skills.

It needs to be noted that the way knowledge is presented to the user is very important. This is a separate issue from that of being able to adjust the system's teaching strategies according to the requirements of a particular body of knowledge and user's needs. This issue has not been addressed in this paper but is particularly important in engineering domains where frequently diagrams can capture information more succinctly than text.

REFERENCES

1. R. G. Carter, Engineering curriculum design, *IEE Proc.*, **131A** (9), 678–3 (1984).
2. A. Newell, in D. Tuma and R. F. Reif (eds), *Problem Solving and Education, Erlbaum*, Hillsdale NJ (1980).
3. B. Cox, E. Pollitzer and J. Jenkins, Explaining and understanding engineering problems—an intelligent tutoring approach. In J. S. Gero (ed.), *Artificial Intelligence in Engineering: Diagnosis and Learning*, Elsevier, Amsterdam, pp. 387–401 (1988).
4. B. M. Cox, Reconciling problem-solving and instruction in one knowledge decomposition framework. In N. Shadbolt (ed.), *Research and Development in Expert Systems VI*, Cambridge University Press, Cambridge, pp. 34–42 (1989).
5. B. Cox, An intelligent tutor for engineering education, *Proc. IEEE Conference on Computer Aided Engineering Education*, pp. 195–201 (1993).
6. B. S. Bloom, *Taxonomy of Educational Objectives; Handbook 1—Cognitive Domain*, David McKay, New York (1956).
7. L. Van den Brande, *Flexible and Distance Learning*, Wiley, New York (1993).
8. E. Wenger, *Artificial Intelligence and Tutoring Systems, Morgan Kaufman*, New York (1987).
9. K.Van Lehn and K. Student, Modelling. In M. C. Polson and J. J. Richardson (eds), *Foundations of Intelligent Tutoring Systems*, Erlbaum, Hillsdale, NJ (1987).
10. J. S. Brown and K. Van Lehn, Repair theory: a generative theory of bugs in procedural skills. *Cognit. Sci.*, **4**, 379–426 (1980).
11. B. Cox, Communicating conceptual integrity in distributed systems through intelligent assistance, *Omega*, **22** (2), May (1994).

Benita M. Cox is a lecturer in information management at Imperial College of Science, Technology & Medicine. She holds an M.Sc in management science and a Ph.D. in artificial

intelligence from Imperial College. She has worked in the area of intelligent tutoring systems on three EC-sponsored Esprit projects and has published widely in this area. Her research interests include computerized simulation and a joint paper in this area with Jeremy Hall recently won the best research paper at the 1994 Absel Conference in San Diego. Her lecturing and consulting interests include the field of information management.