

Knowledge Types and Tutoring Interactions in an Engineering ITS

Ruddy Lelouche & Jean-François Morin

*Département d'informatique, Université Laval, Québec, G1K 7P4,
Canada, Tel.: (418) 656-2131, ext. 2597, Fax: (418) 656-2324,
E-mail: {lelouche, morin}@ift.ulaval.ca*

Abstract

In research on intelligent tutoring systems, the advantage of separating domain knowledge from tutoring knowledge has been proven years ago. However, many scientific disciplines, and in particular all engineering domains, involve not only theoretical knowledge, but also practical one, i.e. problem solving, which consists of practical instantiations of the taught domain theory. In this context, our communication deals with knowledge modelling issues in engineering domains. Firstly, we present a new trichotomic division of the ITS knowledge: domain, problem-solving, and tutoring knowledge. Next, we present a typology of operating modes that can be supported by such a system. Finally, we describe the relationships between these operating modes and the types of knowledge involved in an engineering ITS. Such a division of the system knowledge should facilitate the modelling of tutorial interactions between the student and the system and, therefore, should help the student focus on his own misconceptions and difficulties and optimize his learning.

1 Introduction

Besides its complexity, teaching is as diversified as the variety of teachable domains and that of learning profiles. In particular, teachable domains can be classified according to the type of knowledge to be acquired by the student: “know”, “know-how”, and “know-how-to-be”. Examples of such knowledge types are respectively: anatomy or a language grammar, the skill to solve a mathematical or medical problem, and the capability to adapt to one’s environment or to deal with personal relationships. Knowledge in virtually all engineering domains, and in particular in cost engineering, is of the second type.

Types of knowledge involved in an ITS vary with the type of domain to be taught. In particular, most engineering domains, because they involve problem-solving processes, need knowledge of two types:

- the “know” knowledge, which consists of theoretical knowledge and is modelled in the system using usual knowledge representation techniques like concepts and relations (Brodie & al. [1]), facts and rules (Clocksin & Mellish [6]), semantic networks, etc.;
- the “know-how” knowledge, which consists of practical knowledge, that is, all inferential processes that may be used to solve problems.

In a tutoring system aimed at helping students in engineering, and more generally students in a problem-solving domain, the knowledge to be represented by the system and mastered by the student is thus connected with both knowledge representation (static “know” knowledge) and with reasoning modelling (dynamic “know-how” knowledge), two fundamental domains in artificial intelligence.

For these reasons, problem-solving or know-how domains are the ones in which we are interested here. For the sake of brevity, we denote such a domain a KH-domain. Besides, we deal only with ITSs the domain of which is a KH-domain; we call such an ITS a KH-ITS.

This paper attempts to define and show a possible organisation of the knowledge involved in a KH-ITS. The major interest of such an endeavour is to subdivide knowledge according to the breadth of its applicability. Indeed, many problem-solving techniques are more general than the specific domain in which they are put to use, for example, and can then be easily adapted to other domains (Goldstein [11], Clancey [4], Matsuda & Okamoto [16], Leman et al. [15]).

Also, it has been shown for a long time that, even though expert knowledge about the domain is necessary in any useful ITS (Self [18], Brown [2]), there also exist domain-independent approaches to teaching, i.e. tutoring strategies (Brown & Burton [3], Clancey [4], [5]), that have more to do with the human learning processes and profiles, i.e. the learner’s psychology, than with the domain itself. As with expert systems in general, the problem will be to make these knowledge types communicate with each other (Wenger [19]) independently of how the knowledge itself is organised within each type or module, i.e. to define the communication paths between them. This problem is indeed at the heart of knowledge engineering.

We first present the three fundamental types of knowledge involved in a KH-ITS (section 2). Next, we present the four fundamental operating modes which a student can choose when starting a tutoring session (section 3). Finally, we describe the relationships between the types of knowledge and the operating modes in a KH-ITS, taking examples in our ITS in cost engineering (section 4).

2 Three types of knowledge

As outlined in the introduction, knowledge involved in a KH-ITS can be divided into three types:

- domain knowledge;
- problem-solving knowledge;
- tutoring knowledge.

2.1 Theoretical knowledge: domain knowledge

The first type of knowledge involved in every ITS, the *domain knowledge* (DK), contains all theoretical aspects of the knowledge to be taught to the student. Although its specific structure can be varied, it includes concepts and relations about the domain (Brodie & al. [1], Lelouche & Morin [14]), rules, facts (Clocksin & Mellish [6]), possible applicability restrictions, etc.

The main system activities which focus on this knowledge type are:

- explanation of concepts and relations in the teaching domain;
- providing the other modules of the ITS (problem-solving and tutoring) with the necessary domain background knowledge.

2.2 Practical knowledge: problem-solving knowledge

The second type of knowledge is specific to KH-domains, henceforth to KH-ITSs. We call it *problem-solving knowledge* (PSK). It contains all inferential processes used to solve an engineering problem resulting from the instantiation of a practical situation based on the domain knowledge (Kowalski [13], Patel & Kinshuk [17]). In other words, to be able to solve a problem, the problem-solving knowledge needs a theoretical background, which can be found in the domain knowledge.

The main system activities centred on this knowledge type are:

- providing the inferential tools for problem solving;
- providing the inferential tools for coaching a student in a problem-solving session.

The main advantage of separating the problem-solving knowledge from the domain knowledge is that it emphasises the distinction between the theoretical and practical aspects of the taught domain, thus simplifying the learning process. That knowledge separation into DK and PSK is common to all KH-domains — in particular, engineering domains — which all involve some skill to solve problems. This is why we believe that a KH-ITS, which is supposed to help the student to learn that skill, should bear the same knowledge separation. Besides, that separation is also at the heart of the definition of four generic operating modes in a KH-ITS, as shown in section 3.

2.3 Tutoring knowledge

Finally, the *tutoring knowledge* (TK) contains all tutoring processes enclosed in the ITS. That knowledge type is not directly related to the teaching domain or to problem solving, but helps the student to understand, assimilate,

and master the knowledge included in DK and PSK ((Brown & Burton [3], Clancey [4], Gagné & al. [10], Gagné & Trudel [9])).

The main system activities using tutoring knowledge are:

- ordering and formatting the topics to be presented to the student;
- monitoring a tutoring session, i.e. triggering the various tutoring processes according to the system tutoring goal and the student's actions; such monitoring may imply giving explanations, asking questions, changing to another type of interaction, etc.;
- in a KH-domain, while the student is solving an exercise, monitoring the student's problem-solving activities: understanding and assessing these activities, giving advice to correct or optimize them, giving hints or partly solving the exercise at hand (as required by the student or by the tutoring module), etc.;
- continuously analysing the student's progress in order to optimize the tutoring process.

The advantage of separating the tutoring knowledge from the knowledge of the domain to be taught has been emphasised long ago (Brown [2], Clancey [4], Wenger [19], etc.), and lies in the reusability of TK in various domains. In the case of KH-domains, the domain to be taught clearly encompasses both DK and PSK; indeed, the term "domain knowledge" should clearly apply to DK if referring to the knowledge type as described here, and to DK + PSK if referring to the general knowledge to be acquired. Therefore, as shown earlier, in a KH-ITS, knowledge ends up being separated into three categories rather than the two classical ones (domain knowledge and tutoring knowledge).

3 Operating modes

Depending on what he wants to do with the system, the student can choose one of two ways of using it: either he wants to learn something new, or he wants to check the correctness of what he thinks he has acquired. In the former case, the knowledge he wants to acquire is obviously transferred from the system to the student. In the latter case, the student provides the system with the knowledge (supposedly acquired) that he wants the system to assess; the involved knowledge is therefore transferred from the student to the system.

Besides, as that of a human tutor in a KH-domain, the objective of a KH-ITS is to help the student master the knowledge in both DK and PSK. Indeed, unlike the knowledge in these first two classes, tutoring knowledge is not to be taught to the student; its only purpose is to help the student to learn more efficiently the components of domain knowledge and problem-solving knowledge. Therefore, for whichever reason the student wants to use the system, the involved knowledge may be either essentially factual and theoretical (DK) or essentially practical and applied to a problem (PSK). As a result, these considerations yield altogether four distinct *generic operating modes*, which we now examine in turn.

3.1 The student wants to learn

Whenever the student wants to learn something new, i.e. to acquire new knowledge, he takes the initiative of the knowledge transfer. That transfer then takes place from the teacher to the student, like in a lecture situation. There are two operating modes satisfying that context, according to the type of knowledge to be acquired: the domain-presentation mode and the demonstration mode.

3.1.1 Domain-presentation mode: domain knowledge

In the first mode, called *domain-presentation mode*, the student asks the system some information about a domain theoretical element, and the system reacts by transferring to the student the required information or knowledge. The knowledge involved in this category is always DK.

3.1.2 Demonstration mode: problem-solving knowledge

In the second mode, called *demonstration mode*, the student asks the system to solve a practical problem, e.g. in engineering, or to coach him while he solves such a problem. In the first case, the problem typically comes from the student himself, whereas in the latter one the problem typically comes from the tutoring system. In either case, the main knowledge type involved is thus PSK.

3.2 The student wants to assess his learning

As an alternative to acquiring new knowledge, the student may wish to assess his learning, or in other words to check the correctness of what he is supposed to already know, i.e. *existing* knowledge (Khuwaja & al. [12]). In this case, the student asks the system to take the initiative. The knowledge transfer then takes place from the student to the teacher, like in an exam situation. As when the student wants to learn, there are two operating modes, depending on the type of knowledge to be verified: the domain-assessment mode and the exercising mode.

3.2.1 Domain-assessment mode: domain knowledge

In the third mode, called *domain-assessment mode*, the system basically prompts the student to develop a domain element and the student thus expresses his understanding of that element. If judged necessary, the system may then intervene to correct that understanding. The knowledge involved in this mode is essentially DK.

3.2.2 Exercising mode: problem-solving knowledge

In the fourth and last operating mode, the *exercising mode*, the system prompts the student to solve a practical problem, e.g. in engineering. The student then solves it step by step, showing what he understands of the involved problem-solving knowledge and of the associated domain knowledge. If neces-

sary, the system may decide to intervene in order to help him reach his goal. The main knowledge involved in this mode is naturally PSK.

4 Relationships between operating modes and tutoring knowledge

As shown above, the four operating modes can be distinguished according to two orthogonal criteria: the student's major reason for using the system, i.e. learning vs. learning assessment, and the type of knowledge involved, i.e. theoretical vs. practical. Like in section 3, we present the relationships of tutoring with the operating modes, primarily according to the former criterion, i.e. according to the direction of the knowledge transfer. Then, for each such direction, we examine in turn the typical tutoring activities naturally associated with either operating mode relevant for that direction, and thereafter the tutoring activities which may depart from these natural ones.

4.1 The student wants to learn

When the student wants to learn, i.e. in the domain-presentation (4.1.1) and demonstration (4.1.2) modes, the tutorial activities are mainly concerned with how the material transmitted to the student is ordered and presented. In each case, various tutoring activities are possible (4.1.3).

4.1.1 Domain-presentation mode

In the domain-presentation mode, the knowledge involved is essentially DK. Indeed, taking any engineering domain as an example, the knowledge presented consists of theoretical concepts and relations, facts and rules, etc., of the taught domain.

In this mode, tutoring knowledge is mainly used to monitor the presentation (order of the topics, way of presenting each element, etc.) and to tailor it according to the student's learning profile and needs.

4.1.2 Demonstration mode

By definition, in the demonstration mode, the knowledge involved is mainly PSK. However, as described in Section 2.2, problem solving cannot be achieved without a theoretical background providing the problem data, which lies within DK.

Tutoring knowledge is mainly used to present a problem or problem step, and to adjust it according to the student's profile and knowledge status, possibly displaying various ways of explaining the solving process, or even using various solving methods.

4.1.3 Pedagogical activities involved in these modes

In these first two modes, the tutoring modules can fulfil various tasks to help the student reach his goal. Some corresponding tutoring activities can be:

- to evaluate whether the actual presentation (of DK in domain-presentation mode; of PSK in demonstration mode) made by the system is or is not adequate and sufficient, and to act accordingly;
- to focus on a given concept or relation with which the student needs help (understanding it in domain-presentation mode; adequately using it in demonstration mode);
- to insist on a given point and present it in an alternate way to make sure that the student finally understands.

While involved in such tutoring activities, the system may, and is likely to, temporarily use another type of interaction than the one privileged in the current operating mode, in order to better fulfil its pedagogical mission. Two levels of such *mode shifts* may be defined:

- a first-level shift is a temporary change in the interaction type, which keeps the same knowledge-transfer direction as that indicated by the current mode; that is, in the domain-presentation mode, the system may complement a theoretical presentation with a solved exercise or, conversely, in the demonstration mode, it may recall the involved theoretical elements while presenting a solved problem;
- a second-level shift may or may not change the interaction type, but always changes the direction of the knowledge transfer: in order to assess his comprehension of a presented domain element (domain-presentation mode) or of a solved exercise (demonstration mode), the system may choose either to give the student an exercise to solve or to ask him a theoretical question.

4.2 The student wants to assess his learning

When the student wants to assess his learning, i.e. in the domain-assessment (4.2.1) and exercising (4.2.2) modes, the tutorial activities are mainly concerned with the assurance that the student properly understands the domain knowledge involved and knows how to use it correctly (Khuwaja et al. [12]). As for the modes described in section 4.1, various tutoring activities are possible (4.2.3).

4.2.1 Domain-assessment mode

When the system assesses the student's knowledge with theoretical questions, such questions naturally focus on some DK element(s). Again, the tutoring processes used in TK are those aimed at verifying the student's comprehension.

4.2.2 Exercising mode

When the student asks the system to assess his practical capability to use DK elements in context, i.e. his problem-solving ability, he is given a problem to solve. The involved knowledge is always PSK (still with DK as a theoretical background), for generating the problem as well as for checking the student's

resolution steps. The main role of this mode is to detect misconceptions in the student's problem-solving activities and repair them accordingly.

4.2.3 Pedagogical activities involved in these modes

Some pedagogical activities involved in these two modes are:

- if the student says something wrong, either about DK in domain-assessment mode or about PSK in exercising mode, the system may verify whether this erroneous statement is an occasional slip or a true misconception, by asking him to repeat or reformulate his answer;
- if it is indeed a true misconception, i.e. if the system uncovers an important knowledge element unmastered by the student, it may ask him a subquestion (theoretical or practical as the case may be) focused on the erroneous knowledge element, in order to help him become aware of his misconception and correct it accordingly;
- etc.

As with the tutoring activities presented in section 4.1, the system may temporarily use another type of interaction in order to optimize its tutoring process. Again, two levels of *mode shifts* may be defined, depending on whether or not there is a change in the knowledge transfer direction.

- At the first level, the system still assesses the student's learning, but using interaction types normally associated with the alternate operating mode, thereby changing the knowledge type involved. For example, if the system detects a misconception while the student is solving a problem in the exercising mode, it may ask him a theoretical question in order to help him correct that misconception. Conversely, while in domain-assessment mode, in order to help the student correctly understand the theoretical domain element at hand, it may give him a problem to solve involving that element.
- At the second level, like in 4.1.3, the system may change the direction of the knowledge transfer, as well as the knowledge type. For example, while the student is answering a question (either theoretical in the domain-assessment mode, or practical in the exercising mode), the system may choose to directly correct a detected misconception, either by recalling an involved domain element (an interaction normally associated with the domain-presentation mode) or by solving a (piece of) problem (an interaction normally associated with the demonstration mode), thereby changing the direction of the knowledge transfer.

5 Conclusion

This introduction to know-how ITSs, particularly frequent in engineering, shows why and how, besides the domain knowledge and tutorial knowledge usually found in any ITS, a new type of knowledge needs to be introduced: the *problem-solving knowledge*. While the domain knowledge is useful for all application domains, the problem-solving skills involved in a know-how domain can be demonstrated and exercised only by modules expert in problem

solving, thus involving problem-solving knowledge. In the ITS itself, tutorial knowledge still covers all pedagogical interactions: for theoretical issues, it relies on the domain knowledge like in any other ITS, but for practical issues, it may also use the problem-solving knowledge as an inference engine, like a classical expert system would do.

We also introduced a *trichotomic division of knowledge* into domain knowledge, problem-solving knowledge, and tutorial knowledge. Its major interest is that it subdivides knowledge according to the breadth of its applicability. Like in any ITS, the separation of tutoring knowledge from the knowledge to be taught allows us to hope that domain-independent tutoring strategies can be transported from an ITS to another in a different domain. But in addition, whenever problem solving techniques are more general than the specific domain in which they are put to use, these techniques should be easily adaptable to other know-how domains, thus justifying the division of the knowledge to be acquired into theoretical domain knowledge and practical problem-solving knowledge.

Besides, because of the duality between domain knowledge and problem-solving knowledge, the *four operating modes* that we designed for our ITS in cost engineering may be considered as generic modes for tutor-student interactions in any problem-solving domain, in particular in engineering. Finally, through establishing some relationships between operating modes, knowledge types and tutorial interactions, we showed how the various knowledge types, and mainly how the various tutorial interaction types, can intervene under any operating mode, using adequate *mode shifts* if necessary.

The main advantage of our knowledge modelling as a whole is that it can readily be applied and generalised to all know-how domains, i.e. domains involving some theoretical knowledge, usually represented as concepts, relations, facts, rules, etc., together with know-how-type skills in solving problems, often of a computational and logical nature, operating on that theoretical knowledge. Such domains can obviously be found in all fields of engineering, but also in most exact or near-exact sciences, from mathematics to physics to biology, in medicine (physician's reasoning), and even in some social sciences such as psychology, public relations, politics, or law (judge's or lawyer's reasoning).

Keywords

Artificial intelligence, intelligent tutoring systems, knowledge representation, reasoning modelling, domain knowledge, problem solving, tutoring strategies.

References

1. Brodie, Michael L., John Mylopoulos & Joachim W. Schmidt, eds. (1984) *On Conceptual Modelling, Perspectives from Artificial Intelligence, Databases, and Programming Languages*. Springer Verlag (New York).
2. Brown, John S. (1977) "Uses of artificial intelligence and advanced computer technology in education". *Computers and Communications: Implications for Education* (R. J. Seidel & M. Rubin, eds.). Academic Press (New York), p. 253–270.

3. Brown, John S. & Richard R. Burton (1978) "Multiple representations of knowledge for tutorial reasoning". *Representation and Understanding: Studies in Cognitive Science* (D. G. Bobrow & A. Collins, eds.). Academic Press (New York), p. 311–349.
4. Clancey, William J. (1979) "Dialogue management for rule-based tutorials". *Proceedings of the 6th International Joint Conference in Artificial Intelligence (IJCAI-6)*, p. 155–161.
5. Clancey, William J. (1986) "Qualitative student models". *Annual Reviews of Computer Science* (Traub, J. F., ed.) vol. 1, p. 381–450. Annual Reviews, Inc. (Palo Alto, CA).
6. Clocksin, William F. & Christopher S. Mellish (1981) *Programming in Prolog*. Springer-Verlag (Berlin & New York).
7. Díaz de Ilarraz Sánchez, Arantza & Isabel Fernández de Castro, eds. (1996) *Computer-Aided Learning and Instruction in Science and Engineering*, *Proceedings of the Third International Conference, CALISCE'96*, San Sebastian, España, 29–31 July 1996. LNCS 1108, Springer (Berlin).
8. Frasson, Claude, Gilles Gauthier & Alan Lesgold, eds. (1996) *Intelligent Tutoring Systems*, *Proceedings of the Third International Conference, ITS'96*, Montréal, Canada, 12–14 June 1996. LNCS 1086, Springer (Berlin).
9. Gagné, Denis & André Trudel (1996) "A highly flexible student-driven architecture for computer-based instruction". In [8], p. 66–74.
10. Gagné, Robert M., Leslie J. Briggs & Walter W. Wager (1992) *Principles of Instructional Design*, fourth edition (1st edition published in 1974). Harcourt Brace Jovanovich (Orlando, FL).
11. Goldstein, Ira P. (1977) *The computer as coach: an athletic paradigm for intellectual education*. AI Lab Memo 389, Massachusetts Institute of Technology (Cambridge, MA).
12. Khuwaja, Ramzan, Michel Desmarais & Richard Cheng (1996) "Intelligent Guide: Combining user knowledge assessment with pedagogical guidance". In [8], p. 225–233.
13. Kowalski, R. (1979) *Logic for Problem Solving*. North-Holland (Berlin).
14. Lelouche, Ruddy & Jean-François Morin (1996) "The formula: a relation? Yes, but a concept too!". In [7], p. 176–185.
15. Leman, Stéphane, Pierre Marcenac & Sylvain Giroux (1995) "A multi-agent approach to model student reasoning process". *Proceedings of the 7th World Conference on Artificial Intelligence in Education*, Washington, USA, p. 258–265.
16. Matsuda, Noboru & Toshio Okamoto (1996) "Parallel computing model for problem solver towards ITSs". In [8], p. 234–242.
17. Patel, Ashok & Kinshuk (1996) "Applied artificial intelligence for teaching numeric topics in engineering disciplines". In [7], p. 132–140.
18. Self, John A. (1974) "Student models in computer-aided instruction", *International Journal of Man-Machine Studies IJMMS*, vol. 6, p. 261–276.
19. Wenger, Étienne (1987) *Artificial Intelligence and Tutoring Systems*. Morgan Kaufmann (Los Altos, CA).