Modelling the Knowledge and Reasoning of Users in a Knowledge-Based Authoring Tool

Maria Virvou
Department of Informatics,
University of Piraeus,
Karaoli & Dimitriou 80,
18534, Piraeus, Greece
E-mail: mvirvou@unipi.gr

Abstract: In this paper the domain-independent user modelling capabilities of a knowledge-based authoring tool are described. User modelling is primarily based on a cognitive theory that models human reasoning and is called Human Plausible Reasoning. Moreover the cognitive theory is combined with the overlay modelling technique that aims at modelling the students' knowledge as a subset of the domain knowledge. This method is used in the authoring tool to help the instructors create Intelligent Tutoring Systems (ITS) that may model both the learners' level of factual knowledge on a particular domain and their reasoning abilities within this domain. The resulting ITSs aim at letting the learners justify their answers to questions related to the domain taught. In this way, the system may assess the students' knowledge by taking into account the reasoning that the students have used. On the other hand, the students have the opportunity to consolidate what they have learned by elaborating on their knowledge. At the user interface level, students are able to work on questions in one of two different modes: the tutor mode and the co-learner mode.

Keywords: Authoring tools, intelligent tutoring systems, student modelling, animated agents.

Biographical notes: Maria Virvou is an assistant professor in the Department of Informatics at the University of Piraeus, Greece. She has received her Ph.D. in Artificial Intelligence and Computer Science at the University of Sussex. She has received an M.Sc. degree in Computer Science from University College London. Her first degree in Mathematics was obtained from the University of Athens, Greece. Her current research interests are in user modelling, adaptive and intelligent user interfaces, knowledge-based software engineering, object-oriented software engineering, artificial intelligence in education and e-learning.

1. Introduction

Computer-based education is a fast-growing field that has benefited a lot from the advances of computer technology. However, many researchers (e.g. Salomon, 1990; Welch & Brownell, 2000) point out that technology is effective when developers thoughtfully consider the merit and limitations of a particular application while employing effective pedagogical practices to achieve a specific objective. For example, Hasebrook and Gremm (1999) note that multimedia could potentially facilitate learning processes but they also point out that it has been argued that learning gains are due to instructional methods; therefore many researchers aim to make their multimedia systems more effective using "intelligent" software technologies to adapt to the learner's demands, abilities and knowledge.

Adaptivity to individual learners may be achieved by Intelligent Tutoring Systems (ITSs). As Self (1999) points out, ITS research is the only part of the general information technology and education that has as its scientific goal to make computationally precise and explicit forms of educational, psychological and social knowledge, which are often left implicit. Indeed, ITSs are quite good at providing dynamic aspects to the reasoning ability of educational applications. This is mainly due to their student modelling component that aims at gaining an understanding of how a student learns.
and what the student’s misconceptions may be. Moreover, the advice generator of an ITS also provides dynamic features since it adapts tutoring strategies to individual students.

However, ITSs have often been criticised that they represent immensely complex approaches to building learning environments (e.g. Boyle, 1997). This may also be the reason why ITSs have not been widely used in schools. A solution to the problem of construction of ITSs may be authoring tools that provide the facility of building multiple cost-effective ITSs. Indeed, Murray (1999) highlights the potential of ITS authoring tools in giving the instructional designer a combination of facilities to produce visually appealing, interactive screens and a deep representation of content and pedagogy.

Knowledge-based authoring tools are meant to be used by instructors who wish to author their own ITSs on a certain domain. Therefore, the methods incorporated in the authoring tools have to be domain-independent. On the other hand, the resulting ITSs should have a user modelling capability that helps learners and tutors diagnose the learners’ weaknesses so that remedy is provided. These requirements render the construction of authoring tools difficult. One of the main difficulties is the high degree of generality and domain-independence that has to be achieved while at the same time there has to be effectiveness and user-friendliness for the potential authors and effectiveness and adaptivity for the students who will use the end result.

In this paper a domain-independent method is described for providing user modelling capabilities to an ITS authoring tool, a small part of which has also been described in (Virvou, 2000). The domain-independent method is primarily based on an adaptation of a cognitive theory that aims to model human reasoning and is called Human Plausible Reasoning (Collins & Michalski, 1989), henceforth referred to as HPR. The theory formalises the plausible inferences based on similarities, dissimilarities, generalisations and specialisations that people often use to make plausible guesses about matters that they know partially. These inferences may lead to either correct or incorrect guesses; in any case these guesses are plausible. HPR has been adapted and used previously in learning environments for novice users of UNIX (Virvou, 1999; Virvou & Du Boulay, 1999) and for novice users of a Graphical User Interface (Virvou & Kabassi, 2000; 2001). However, it has not been previously used in an authoring tool although it provides a promising tool for the domain-independent reasoning required for user modelling.

The incorporation and adaptation of this theory in the authoring tool aims at providing environments where students will be able to answer questions and justify their reasoning. In this sense the system constructs environments where there is opportunity for a negotiating teaching-learning dialogue between the ITS and the students. Collaborative discourse is an issue that has attracted a lot of research energy in the recent years (e.g. Moore, 2000; Hume, Michael, Kahler, Barlow, Stone & Bhogal 1996; Baker, 1994; Fox, 1993). If designers of future tutoring systems wish to capitalise on the knowledge gained from human tutoring studies, the next generation of tutoring systems will incorporate pedagogical agents that engage in learning dialogues (Person, Grass, Kreuz, Pomeroy & the Tutoring Research Group, 2001).

In addition to HPR, the overlay technique has been used to model the state of the students’ knowledge. In this way, the user modelling component of the authoring tool is based on a novel combination of methods that complement each other and thus the resulting ITSs may both model the reasoning of users, their state of knowledge of the domain and possible misconceptions they may have about the domain being taught.

The remainder of this paper is organised as follows: Section 2 presents a summary of HPR, which has been adapted and used as the primary reasoning mechanism for user modelling. Section 3 describes the authoring procedure for potential authors of the authoring tool. Sections 4, 5 and 6 describe the ITSs that may be created by using the authoring tool. Finally, Section 6 presents the conclusions drawn from this work.

2. Human Plausible Reasoning Theory

The core theory consists of:

a) A set of primitives.

A summary of the primitives is given in Table 1.

The primitives can be classified into four groups:

a) Statements representing people’s beliefs about the world.

b) Statements involving relations (i.e. GEN, SPEC, SIM, DIS).

These represent different relationships among concepts in hierarchies.

c) Relational expressions, which are either mutual implications or mutual dependencies.

These represent people’s approximate knowledge about what depends on what, which can be specified with more or less precision.
d) **Certainty parameters** that act to condition these three kinds of expression and which affect the certainty of the different inferences described in the next two sections.

The primitives consist of:

a) Basic expressions:
   i. Statements.
   ii. Relational expressions.

b) Operators.

c) Certainty parameters

b) A set of inference rules.

a) **Statement transforms**.

b) **Transforms based on dependencies and implications.**

<table>
<thead>
<tr>
<th>arguments $\alpha_1, \alpha_2, f(\alpha_1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g. Sam, whale, Sam’s food</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>descriptors $d_1, d_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g. size, animal-type</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>terms $d_1(\alpha_1), d_2(\alpha_2), d_2(d_1(\alpha_1))$</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g. animal-type(Sam), size(whale), size(animal-type(Sam))</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>referents $r_1, r_2, {r_2\ldots}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g. whale, large, large plus other sizes.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Statements $d_1(\alpha_1) = r_1 : \gamma, \phi$</th>
</tr>
</thead>
<tbody>
<tr>
<td>e.g. size(whale)=large: certain, high frequency</td>
</tr>
<tr>
<td>(translation: I am certain almost all whales are large)</td>
</tr>
</tbody>
</table>

**Table 1: HPR’s elements of expressions**

The simplest class of inference are called **statement transforms**. If a person believes some statement such as that the flowers growing in England include daffodils and roses which translates to flower-type(England) = daffodils, roses,… there are eight statement transforms which allow plausible conclusions to be drawn. The argument transforms move up, down or sideways in the argument hierarchy using GEN, SPEC, SIM or DIS respectively. The **referent transforms** do the same in the referent hierarchy. For example, from the statement flower-type(England)=roses, one can make the following statement transforms, given a type hierarchy for flowers and a similar type hierarchy for geographic regions (not illustrated).

**Argument transforms**

- GEN flower-type(Europe)=roses
- SPEC flower-type(Surrey)=roses
- SIM flower-type(Holland)=roses
- DIS flower-type(Brazil)≠roses

**Referent transforms**

- GEN flower-type(England)=temperate flowers
- SPEC flower-type(England)=yellow roses
- SIM flower-type(England)=peonies
- DIS flower-type(England)≠bougainvillea

The formal representation of the similarity statement transforms, which are quite important is the following:

**SIM-based argument transforms**

\[
d(a) = r: \gamma_1, \phi, \mu_\alpha \quad a \text{ SIM} a \text{ in}\ CX(A, D(A)): \sigma, \gamma_2
\]

\[
D(A) \leftrightarrow d(A): \alpha, \gamma_3
\]

A, a \text{ SPEC} A: \gamma_4, \gamma_5

\[
d(a') = r: \gamma = f(\gamma_1, \phi, \mu_\alpha, \sigma, \gamma_2, \alpha, \gamma_3, \gamma_4, \gamma_5)
\]

E.g. livestock(West Texas)=cattle,…: $\gamma_1 = high, \phi = high, \mu_\alpha = high$

Chaco SIM West Texas in CX(region, vegetation(region)): $\phi = moderate, \gamma_2 = moderate$
vegetation(region) ↔ livestock(region): $\alpha = \text{high}, \gamma_3 = \text{high}$

West Texas, Chaco SPEC region: $\gamma_4 = \text{high}, \gamma_5 = \text{high}$

livestock(Chaco) = cattle,…: $\gamma = \text{moderate}$

SIM-based referent transforms

$d(a) = r$,…: $\gamma_1, \phi, \mu_r$
$r' \text{ SIM } r \text{ in } CX(d, D(d)): \sigma, \gamma_2$

$D(d) \leftrightarrow A(d): \alpha, \gamma_3$

a SPEC A: $\gamma_4$

$d(a) = r': \gamma = f(\gamma_1, \phi, \mu_r, \sigma, \gamma_2, \alpha, \gamma_3, \gamma_4, \gamma_5)$

E.g. Sound(wolf)=howl,…: $\gamma_1 = \text{high}, \phi = \text{high}, \mu_r = \text{low}$

Bark SIM howl in CX(sound, means of production(sound)):
$\sigma = \text{high}, \gamma_4 = \text{high}$

Sound(wolf) = bark,…: $\gamma = \text{moderate}$

Human Plausible Reasoning has been applied in the authoring tool as an underlying reasoning mechanism that simulates learners’ reasoning when they answer questions and they do not have a ready answer. Statements are used to model what the learner knows and statement transforms are used to show what possible errors s/he may make. Certainty parameters are used to represent the degree of confidence of the system concerning the inference beliefs.

3. Authoring Procedure

Instructors who may wish to use the authoring tool to produce their own ITS, have to provide information to the system. The information required relates to the domain knowledge, construction of tests and specification of the way that students are going to be marked in tests.

3.1 Description of the domain knowledge

The initial input to the authoring tool is a description of knowledge concerning a specific domain given by a human teacher who is acting as an author. At first, the domain has to be described in terms of hierarchies, which constitute the knowledge representation of HPR. Therefore the author has to decide what the main concepts of the lesson are, that may be represented in hierarchies. Then s/he may create hierarchies by giving data to a dialogue box of the system. After this input has been given, the tool constructs a knowledge base concerning the specific domain in the form of hierarchies. Then the author inserts facts that s/he wishes to be taught to students and which are relevant to the main concepts of the hierarchies. Finally, the authoring tool may assist the author to construct tests that consist of questions relating to the factual knowledge of the domain.

![Figure 1. Creation of hierarchies](image)
For example, an author wishes to create a lesson in geography about fruits in the countries of Europe. S/he may create the main hierarchy that represents the relations of countries as the author perceives them by using a dialogue box as shown in the example of Figure 1. Then s/he may create a secondary hierarchy that represents relations of fruits. An example of a main hierarchy and a secondary hierarchy is shown in Figure 2, which illustrates a small part of a country hierarchy and a small part of a fruit hierarchy.

![Figure 2. Examples of parts of hierarchies](image)

The author then has to insert certain values for parameters, which characterise the relations among objects of the hierarchies. Such parameters include the degree of typicality of a child-node in relation to its parent-node and the degree of similarity among nodes of the same level. For example, the instructor may state that the typicality of lemons in the category of citrus fruits is 0.8 (the value of the degree ranges between 0 and 1). This means that the instructor states that the lemon is quite typical of the category where it belongs.

After the author completes the insertion of the necessary data about the hierarchies, s/he enters the facts that s/he wishes to teach to students in the particular lesson. Examples of facts may be the following:

- Citrus fruits are produced in Greece.
- Oranges are produced in Spain.
- Apples are produced in Italy etc.

All of the above facts constitute the knowledge-base of the ITS that students may use. In particular, the knowledge-base will be used for the tutoring of students and for examining their knowledge and reasoning abilities within the domain.

### 3.2 Construction of tests and specification of students’ grading

The authoring tool may assist the author to construct tests consisting of questions, which are relevant to the factual knowledge that has been inserted. For example, a question could be: “Does Italy produce citrus fruits?” In every case the questions combine one node of the main hierarchy and one node of a secondary hierarchy.

In particular, if an author wishes to create a new test, s/he selects “New Test” and has to give a unique name to the new test. Then s/he has to select the section of the knowledge base, which is going to be used for the creation of the new test. Then there are going to be two drop-down lists, which the author may use for the creation of each question of the test. The first drop-down list contains the nodes of the main hierarchy of the selected section and the second drop-down list contains the nodes of a secondary hierarchy.

Finally, the author is also required to select a way that learners are going to be marked in tests. In particular, authors are required to state the exact percentage of the grade that students are going to receive for each category of error that the system may recognise as will be explained in more detail in the next section.
4. Overview of the Resulting ITSs

The resulting ITSs may be used by students who can be shown facts from the knowledge base. In addition, students may test their knowledge by answering the questions that the authoring tool has formed, based on the data given by the author. The student is asked to give an answer and then is also asked to give a justification for this answer.

4.1 Negotiating knowledge

When a student is asked a question by the ITS, s/he is expected to give an answer in one of two possible ways:

a) from immediate knowledge or
b) from a guess.

An example of how the student selects to answer questions is illustrated in Figure 3.

![Figure 3](image)

Figure 3. The student's answer may be given "from immediate knowledge" or "from a guess"

The first way should be selected from a learner who is confident that s/he knows the answer. In this case the student has to provide a simple answer to the question asked. The second way should be selected from a learner who does not have an immediate answer to the question asked and wishes to make a guess. In this case the student has to provide both an answer and a justification for this answer.

For example, in the ITS for geography that will have resulted from the above authoring example, the student may be asked the following question: “Does Greece produce oranges?” Then the student should select the way that s/he is going to answer the question. In any case the student may give either a positive or negative answer. If the student believes that s/he knows the answer, s/he gives a straight answer of the type: “Yes” or “No”. In this case, the justification that is implied is “because I know so”. If the student does not have a ready answer because s/he may not possess the appropriate piece of factual knowledge for this answer, s/he is asked to make a plausible guess and justify it. For example, the student may answer: “My guess is yes. I know that Italy produces oranges; Italy is similar to Greece in terms of agricultural products. Therefore, there is a high possibility that oranges are produced in Greece as well.” The student’s guess may be correct or incorrect. However, the reasoning that s/he has used may reveal whether the student has a good knowledge of geography and whether s/he is able to use it plausibly.

The correctness of immediate answers, guesses and/or justifications are recorded to the long term student model. By the end of the test, the system can have a good idea of what the student has learned and how well s/he uses this information to answer questions that s/he may not know precisely. In particular, there may be several cases of combinations in terms of the correctness of a guessed answer and the correctness of the justification:

1. Both the guessed answer and the justification are correct.
2. The guessed answer is wrong but the reasoning in the justification is correct. In this case the system informs the student about the mistake but admits that the reasoning was plausible.
3. The guessed answer is correct but the justification is incorrect. In this case the system informs the student that the answer was correct by chance, therefore the student does not get any credit for this answer.
4. Both the guessed answer and the justification are incorrect.

For the cases 3 and 4 where the justification is incorrect, there are two possibilities as to the reason why the justification is incorrect:

a) The justification is incorrect because the reasoning has been completely incorrect.
b) The reasoning is partly correct but the factual knowledge of relevant topics that has been used is incorrect. For example, if the student was asked whether Greece produced pine apples and the student answered: “My guess is yes because pine apples are produced in Italy and Italy is similar to Greece in terms of agricultural products” the student’s reasoning would have been based on incorrect facts. In fact Greece is similar to Italy but Italy does not produce pine-apples.

In this way, for each question, the ITS takes into account a broader part of the student’s knowledge and skills than appears in the question alone. Indeed, as Andriessen and Sandberg (1999) point out, the process of becoming an expert in a certain domain should no longer be solely viewed as the acquisition of a representation of correct knowledge; the knowledge to be acquired should flexibly manage open problems. In addition, Ohlsson (1993) suggests the analysis of epistemic activities (arguing, describing, explaining, predicting, etc.) to be more relevant for higher order learning of declarative knowledge than the study of goal-oriented action.

4.2 Architecture of the ITSs

The architecture of the ITSs that will be generated by the authoring tool follows the main line of the architectures of ITSs. It has been widely agreed that an ITS should consist of four components, namely the domain knowledge, the student modeller, the tutoring component and the user interface (Hartley & Sleeman, 1973; Wenger, 1987; Self, 1999). The domain knowledge consists of a representation of the domain to be taught (e.g. Biology, Chemistry, etc.). The student modeller constructs a model of the student in terms of her/his knowledge level and her/his problem-solving performance. The tutoring component contains a representation of the teaching strategies of the system. Finally the user interface is responsible for the communication with the student.

In the ITSs that are generated by the authoring tool, the student modelling component examines the correctness of the students’ answers in terms of the students’ factual knowledge and reasoning that they have used. The reasoning of the student is examined by the HPR Modellerer, which is the component that is based on an adaptation of Human Plausible Reasoning. The state of the student’s knowledge of the domain is examined by the Overlay Modeller, which is the sub-component of the student modelling that operates based on the overlay technique (Stansfield, Carr & Goldstein 1976). Information about each student concerning his/her knowledge and reasoning ability is recorded in his/her long term student model. The long term student model (Rich, 1983) keeps a history record of the student and is updated every time the student answers a question of a test. The student model is used to adapt the presentation of lessons to the particular student’s knowledge and possible weaknesses. The adaptive presentation is performed by the Tutoring Component, which is responsible of showing to the student the parts of the theory that s/he does not know or that s/he wants to read about.

Finally the student interface presents the system’s advice to the user in two different modes, the tutor mode and the co-learner mode. Each of these modes employs an animated agent to communicate with the student and give advice. The architecture of the ITSs that may be produced by the authoring tool is illustrated in Figure 4.

![Figure 4. Architecture of resulting ITSs](image-url)
5. Student Modelling

Student modelling in the ITSs, which result from the authoring tool, is based on HPR, which is combined with the overlay technique. The overlay model was invented by Stansfield, Carr and Goldstein (1976) and has been used in many early user modelling systems (e.g. Goldstein, 1982) and more recent systems (e.g. Matthews, Pharr, Biswas & Neelakandan, 2000). The main assumption underlying the overlay model is that a user may have incomplete but correct knowledge of the domain. Therefore, the user model may be constructed as a subset of the domain knowledge. This subset represents the user’s partial knowledge of a domain and thus the system may know which parts of the theory the user knows and which s/he does not know.

However, as Rivers points out (1989), overlay models are inadequate for sophisticated modelling because they do not take into account the way users make inferences, how they integrate new knowledge with knowledge they already have or how their own representational structures change with learning. To this end, if HPR is combined with the overlay technique, it may provide the reasoning that is missing from the overlay technique. In particular, HPR may provide a domain-independent inference mechanism for modelling the users’ reasoning in relation to the knowledge that these users already have. On the other hand, the overlay technique is useful for the representation of the learner’s state of knowledge which in turn may be used by the system for individualized presentations of the contents of lessons according to the learner’s needs.

Indeed, HPR is compatible with the overlay technique and may be used in conjunction with it. Similarly to the overlay technique, HPR also assumes that a user may have incomplete domain knowledge. In particular, the theory assumes an all or nothing view of a person’s knowledge, meaning that a person is assumed either to have or not to have a piece of knowledge. In this sense, HPR is not directly concerned with misconceptions or incorrect beliefs because it focuses on the inferential process that leads people to derive a belief based on another belief, irrespective of the correctness of either belief. However, the cases where a learner uses this reasoning to produce an incorrect answer reveals both the incorrectness of the belief and the underlying cause of the error. Moreover, the justification of a guessed answer (irrespective of the guessed answer’s correctness) reveals other parts of the user’s knowledge, which are relevant to the question asked and have been used by the user; therefore it provides data for the overlay model.

For example, in an ITS for zoology a student may be asked the following question: “Do ducks have broad wings?” If the student does not have a ready answer because s/he may not possess the appropriate piece of factual knowledge for this answer, s/he may answer: “My guess is yes. I know that geese have broad wings; geese are similar to ducks with respect to their characteristics. Therefore, it is likely that ducks have broad wings too.” In this case, the data that is going to be used for the overlay model consists of the following correct facts that the user has shown evidence of knowing:

1. Geese have broad wings.
2. Ducks are similar to geese.

More specifically, the student modelling component analyses the students’ answers to questions of tests by using two sub-components as illustrated in Figure 4, the HPR modeller and the overlay modeller. If the student selects to answer a question from “immediate knowledge” and the answer is correct then the overlay modeller updates the long term student model by adding the particular piece of knowledge to the known ones by the student. However, if the answer is incorrect then the HPR modeller is activated so that it may perform error diagnosis and find out what the cause of the error has been. For example, in a geography test, if the user has stated that the population of Greece is 15,000,000 instead of 11,000,000, which is correct, the user’s answer is closer to the correct one than a possible answer of 50,000,000. The sizes of the numbers (in the student’s answer and in the correct one) are similar, therefore they could have been confused by the student. After the HPR modeller performs error diagnosis, the overlay modeller updates the model of the student’s knowledge by entering either no knowledge or partial knowledge of the particular fact involved in the question.

In case a student selects to answer a question “from a guess” then s/he will be required to provide an explicit justification of his/her answer. Therefore the justification is passed to the HPR modeller that examines its plausibility in terms of the reasoning used and the correctness of the underlying facts. Then the overlay modeller updates the model of the student’s knowledge concerning the underlying facts that the user has used. Both the HPR modeller and the overlay modeller give information to be recorded in the long term student model.
6. Interaction with the student

The questions that assess the students’ knowledge and reasoning skills, may be answered by a student when s/he is working in one of two different modes, the tutor mode or the co-learner mode. The underlying reasoning used for both modes is the same and is based on HPR. The difference of these modes is restricted to the interface level in a similar way as in another authoring tool for algebra-related domains (Virvou & Moundridou, 2000; 2001). In particular, in the tutor mode, there is an animated agent figuring a tutor that provides rather formal messages and is responsible for assigning a grade to the student whereas in the co-learner mode, there is an animated agent figuring a peer that provides messages in a more casual way. Hence the same conclusions based on the student modelling component may be expressed in two different messages that convey the same kind of advice. For example, in a case where a student is encouraged to read more about Greece the messages are expressed in the following way by the two different animated agents:

- Agent figuring the tutor: “Please read more about Greece. Your knowledge concerning this chapter is incomplete.”
- Agent figuring a co-learner: “I think you have to read more about Greece. You do not seem to be doing well in questions about it.”

The mode of the simulated co-learner has been considered quite important by many researchers for the purpose of improving the educational benefit of tutoring systems. One reason for this is the fact that the simulated student can be simultaneously an expert and a co-learner and thus it can scaffold and guide the human’s learning in subtle ways (Van Lehn, Ohlsson & Nason, 1994). In the present research, the co-learner may scaffold the student’s learning and reasoning by using HPR to reason about the student’s answers in the domain.

Furthermore, the presence of an animated speaking character in the interface of an educational application attracts the students’ attention and thus increases their engagement and motivation. Indeed, previous studies (Lester, 1997; Moundridou & Virvou, 2002) have shown that the presence of a life-like character in an interactive learning environment – even one that is not expressive – can have a strong positive effect on a student’s perception of the learning experience”. When these life-like characters are combined with an underlying kind of reasoning that is based on human plausible reasoning then the interaction may become more human-like and appealing to students than it currently is for many systems.

7. Conclusions

This paper has described an ITS authoring tool that incorporates a reasoning mechanism based on HPR. The reasoning mechanism is used for student modelling which may assess both the student’s factual knowledge on a certain domain and his/her ability to reason within this domain. Moreover the inferential ability of HPR is combined with the overlay technique, which may represent the state of the students’ knowledge as a subset of the domain knowledge. The domain-independence of the theory has allowed its application as the main underlying reasoning mechanism in the authoring tool that required domain-independent inference mechanisms.

The learning environments that result from this authoring tool are quite open for the students to test their factual knowledge on a certain topic and possibly relevant topics and the reasoning they use to deal with open problems by combining factual knowledge from relevant topics. This is quite important for creating environments where users may work on their knowledge and develop skills for arguing, explaining and predicting using their knowledge. This would help them consolidate their existing knowledge, expand it with new knowledge and develop their reasoning skills, which would be useful for any kind of knowledge in any kind of domain.

References


