Maintaining a Jointly Constructed Student Model

Vania Dimitrova, John Self, Paul Brna

Computer Based Learning Unit, Leeds University, Leeds LS2 9NA, UK
e-mail: {V.Dimitrova,J.A.Self,P.Brna}@cbl.leeds.ac.uk

Abstract. Allowing the student to have some control over the diagnosis inspecting and changing the model the system has made of him is a feasible approach in student modelling which tracks the dynamics of student behaviour and provides for reflective learning. We present an approach for maintaining the student model in interactive diagnosis where a computer and a student discuss about the student's knowledge. A belief modal operator is adapted to model the knowledge of the learner and to help in maintaining the interaction between the computer system and the learner. A mechanism for finding agreements and conflicts between system and learner's views is described.

1 Introduction

Modelling a learner's cognitive capacity is essential for an intelligent tutoring system to provide individualised instruction and adaptive interaction [1]. Allowing the student to have some control over the diagnosis and to inspect the model the system has made of him is a feasible approach in student modelling [4] which tracks the dynamics of student behaviour [2] and provides for reflective learning [3]. A similar method is applicable to user modelling [5] and building adaptive systems [6]. A constructive interaction guided by the system where both the computer and the learner reflect on the learner's beliefs is the means for involving the user in diagnosis. Designing such an interactive process needs a dialogue management framework [7] and a formal engine to maintain a student model which is jointly constructed by the system and the learner and accumulates their views about the learner's knowledge.

There are few attempts to formalise the process of maintaining the user/learner model when open for inspection and change directly by the user/learner, see [8,9]. In these projects, the notion of the interaction is very constrained and the formalisations they offer do not consider modelling the process of *interactive reflection* which results in a jointly constructed student model. Such a task is addressed in this paper. We present an approach to formalising the process of maintaining a jointly constructed learner model in interactive diagnosis. The kernel of this approach is a mechanism for finding agreed beliefs and conflicts between the computer system and the learner when discussing the learner's knowledge. We have employed an epistemic operator *belief* in a dialogue game interaction model and have adapted formal specifications from [10] into an interactive diagnostic context.

The conception of a jointly constructed student model can be related to the notion of common and distributed knowledge in multi-agent systems where sound and complete axiomatisations have been provided [11]. However, there is a debatable rationale for adopting strong deductive approaches for inherent problems with computational complexity and natural plausibility in modelling human reasoning [12]. A basic assumption in interactive diagnosis is that the belief set the computer system has about the learner is not complete. Moreover, while the learner is expected to reflect on his knowledge, he may well apply unsound and incomplete reasoning, which the system will seek to correct. Therefore, to model interactive diagnostic situations we have been able to adopt some simplifications.

Several belief models that employ nonmonotonic and limited reasoning have been developed to model agents' beliefs in dialogue simulations (c.f. [12], [13], [14]). Mutual beliefs of the system and the user which these systems consider, particularly mutual beliefs about the user's domain beliefs, are similar to the notion of agreed beliefs in interactive diagnostic dialogue. The agreements play a crucial role in interactive diagnosis presenting the jointly constructed student model. Hence, they have been elaborated in our formalisation, so that not only explicit agreements but also implicit and assumed ones have been modelled. In addition, we define conflicts between the system's and the learner's views about the learner's knowledge which are sources for a negotiative dialogue in interactive diagnosis. This notion of conflicts is different from conflicts between the system's and the user's beliefs which user modelling frameworks use to define the user's erroneous and incomplete knowledge [14]. In our model, the correctness of the learner's beliefs is assessed by comparing the agreements about the student's beliefs with the system's domain knowledge.

Next in the paper, there is a discussion about the process of maintaining the student model in interactive diagnosis and the need for a mechanism for finding agreements and conflicts between the computer and the learner. Such a mechanism will be presented in section 3 and an example of its application will be elaborated in section 4. Finally, further applications of our approach will be discussed.

2 Maintaining the Student Model in Interactive Diagnosis

We use the term *interactive diagnosis* to emphasise the view of diagnosis as an *interactive process* involving two agents, a diagnoser and diagnosee, who discuss and construct together the student model. Open learner modelling environments where the learner is involved in a discussion reflecting upon the model the computer has built of him [4] present interactive diagnostic situations. Other cases are peer diagnosis systems [15] where users reflect on the models they have built about their peers.

2.1 An example

The example below, which will be further elaborated in section 4, is from a session with STyLE-OLM[1] – an open learner modelling component in a terminology learning

[1] STYLE-OLM is an open learner modelling component in the STYLE environment developed within the LARFLAST project: http://www.larflast.bas.bg/. The example here is from a version of STyLE-OLM in a Computer Science domain.

environment [16]. The interaction in STYLE-OLM is grounded on a dialogue maintenance framework [7], the communication language is based on conceptual graphs [17] (for simplicity here we consider propositions stated in a natural language), the domain knowledge base is encoded with conceptual graphs [18].

Initially, the learner model includes learner's beliefs "VISUAL BASIC is an OBJECT-ORIENTED LANGUAGE", "VISUAL C++ is an OBJECT-ORIENTED LANGUAGE", and "OBJECT-ORIENTED LANGUAGE contains OBJECTS". The dialogue strategy of C is to discover a possible misconception behind the learner's erroneous belief that VISUAL BASIC is an OBJECT-ORIENTED LANGUAGE.

(1) C. You think that VISUAL BASIC is an OBJECT-ORIENTED LANGUAGE.
(2) S. I Agree.
(3) C. You think that VISUAL BASIC is a VISUAL LANGUAGE.
(4) S. I Agree.
(5) C. You think VISUAL LANGUAGES are OBJECT-ORIENTED LANGUAGES.
(6) S. I disagree.
(7) C. Do you think that VISUAL C++ is an OBJECT-ORIENTED LANGUAGE.
(8) S. Yes.
(9) C. You think that VISUAL BASIC contains OBJECTS.
(10) S. I agree.
(11) C. Do you think that VISUAL BASIC has a characteristic INHERITANCE?
(12) S. I doubt this.
(13) C. Do you know a relation between OBJECT-ORIENTED LANGUAGE and INHERITANCE?
(14) S. I don't know.
(15) C. You think that VISUAL BASIC is an OBJECT-ORIENTED LANGUAGE BECAUSE it has OBJECTS but OBJECT-ORIENTED LANGUAGES also have a characteristic INHERITANCE which VISUAL BASIC does not have.

The aim of the learner model maintenance is to find out what is to be included in the learner model after the interaction.

2.2 Maintaining the student model in interactive diagnosis - main components

The learner model maintenance process has three main parts (fig.1):
? Ascribing participants beliefs from speech acts;
? Inferring what has been agreed through the interaction;
? Ascribing a level of correctness to learner's beliefs and updating the learner model.

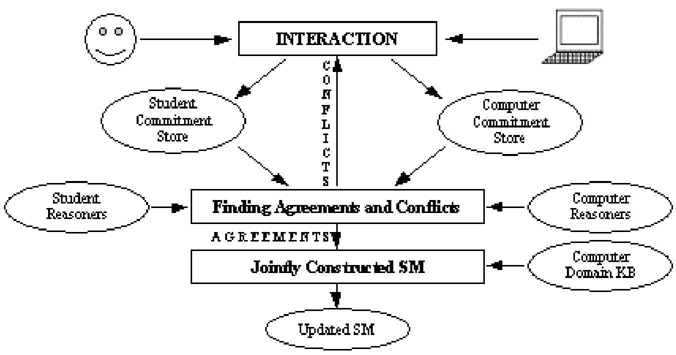

Fig. 1. Maintaining the student model in interactive diagnosis.

We have shown elsewhere that a dialogue management framework based on dialogue games [19] can be adapted for maintaining communication in interactive diagnosis [7]. Following a dialogue game model, commitment rules define the effects of moves upon the dialogue participants' commitment stores. These rules ascribe changes in participants' belief stores after a speech act is uttered. The ascribed beliefs are not added directly to the student model but accumulated in temporarily built commitment stores to keep different viewpoints, which is essential in providing the notion of a collaborative dialogue [3]. To maintain the dynamics of the belief stores, a belief revision approach similar to the one described in [8] is employed.

Being responsible for maintaining the student model which is to be used later by other components of the learning environment, an interactive diagnostic module has to have a mechanism for finding what has been *agreed* through the interaction which is the kernel of a *jointly constructed student model*. After an interaction episode finishes, i.e. the participants agree to change the focus of the discussion, an inference mechanism is to be adopted to refine the beliefs of participants and to find out what these agents have agreed about the learner's beliefs. The agreed beliefs are to be attributed a level of correctness by comparison with the expert's knowledge and then used as a source for updating the learner model. The inference mechanism also has to detect which are the *conflict points* in the participants' belief stores. In the following interactions, these points are to be used as the negotia in a negotiative dialogue game. A possible formalisation of finding agreements and conflicts and maintaining a jointly constructed learner model is described in the rest of the paper.

3 Finding Agreements and Conflicts about the Student's Beliefs

3.1 Assumptions

Our starting point of view is that learner's domain knowledge is modelled in terms of *beliefs* that correspond to the conceptual level of the student model. We consider a structure of the learner model $L : L = B ? R^{mu} ? R^{mc}$ where B is a set of learner's beliefs and consists of correct, erroneous and incomplete beliefs, R^{mu} is a set of misunderstanding rules that represent patterns of potential conversational failures and R^{mc} is a set of misconception rules that define possible reasons for learner's erroneous and incomplete beliefs. We consider R^{mu} and R^{mc} useful for the diagnoser to plan the interaction and B *open* for a discussion. Hereafter, we use two agents a *student s* and a *computer c*.

We will use an epistemic operator $B_s(j)$ to denote that the student believes a fact represented by the propositional formula j. We also use the negation $\neg B_s(j)$ to denote that the student does not believe j (he may not believe $\neg j$ which is $B_s(\neg j)$).

For the computer, expressions $B_c(j)$ and $\neg B_c(j)$ present its domain expertise and are derived from the system knowledge base. We also consider nested beliefs to represent the beliefs that the computer has about the student. Thus $B_c(B_s(j))$ denotes that c believes that s believes j. Its negations are: $B_c(B_s(\neg j))$ c believes that s

believes $\neg j$; $B_c(\neg B_s(j))$ c believes it is not the case that s believes j; $\neg B_c(B_s(j))$ c does not believe that s believes j. Such expressions build the student model in the traditional diagnosis where the computer has the overall control over the diagnostic process. The computer *opens* these beliefs for a discussion with the learner in interactive diagnosis. Such expressions appear in the computer commitment store throughout the dialogue.

Note that learner's beliefs about system's beliefs have been considered redundant. We assume that in the conversation the learner reflects on his beliefs or challenges those from the system's commitment store, e.g. when the learner agrees with $B_c(B_s(j))$, the corresponding commitment rule will add $B_s(j)$ to the learner belief store and when he challenges $B_c(B_s(j))$, $\neg B_s(j)$ will be ascribed.

Before the interaction, we will consider a base set of computer beliefs I_c, a base set of computer beliefs about student beliefs I_{cs}, and a base set of student beliefs I_s. The beliefs in I_c are encoded by a domain expert and those in I_{cs} and I_s are either stated explicitly or assigned by some initial domain specific inference.

3.2 Reasoning

We consider our agents to have a (not necessarily complete) set of *inference rules* that we will call *reasoners*. We will denote the student reasoners over his beliefs by R_s and the computer reasoners over its beliefs by R_c. The computer will also have reasoners about the beliefs of the student - R_{cs}.

The rule $f_1,\ldots,f_n ? f$ will allow us to infer new beliefs from agents' belief sets. For example, $B_c(j_1),\ldots, B_c(j_n) ?_{Rc} B_c(j)$ will allow us to assume that for the computer $B_c(j)$ is *true* if $B_c(j_1),\ldots, B_c(j_n)$ are *true*. The computer might do some inference over its beliefs about the student. For instance, the rule

$$B_c(B_s(j \Rightarrow y)), B_c(B_s(y)) ?_{Rcs} B_c(B_s(j))$$

defines some default reasoning assumptions made by the computer about the learner's reasoning. R_s is the least determined set presenting learner's reasoning over his beliefs. An example of a possible rule from R_s is given below.

$$B_s(j \Rightarrow y), B_s(\neg y) ?_{Rs} \neg B_s(j).$$

We will infer new agents' beliefs by applying their reasoners *only once*. As discussed in [10], there is a certain rationale in adopting limitations. Humans tend not to draw all possible conclusions from their beliefs. Also, it might be considered peculiar to assume that an agent believes a proposition which needs a fairly long inference process upon the agent's beliefs in order to be ascribed as true.

We can now define agent's *belief sets* that will be the essence for deriving agents' agreements and conflicts.

$$B_c = \{f \mid f \in I_c \text{ or } \exists f_1,\ldots,f_n: (f_1,\ldots,f_n ?_{Rc} f) \in R_c,\ f_i \in I_c,\ i = 1\ldots n\}$$

$$B_s = \{f \mid f \in I_s \text{ or } \exists f_1,\ldots,f_n: (f_1,\ldots,f_n ?_{Rs} f) \in R_s,\ f_i \in I_s,\ i = 1\ldots n\}$$

$$B_{cs} = \{f \mid f \in I_{cs} \text{ or } \exists f_1,\ldots,f_n: (f_1,\ldots,f_n ?_{Rcs} f) \in R_{cs},\ f_i \in I_{cs},\ i = 1\ldots n\}$$

3.3 Agreements

A distinguishing characteristic of an interactive diagnostic dialogue is that agents are talking about the beliefs of one of them. This entails a proper adjustment of the mechanism for searching for what has been agreed throughout the interaction. Following a commonsense view that agents agree with something if they have the same opinion about it, we identify the *agreements* in interactive diagnosis as being those *beliefs of the student* that the computer assumes the student believes and the student himself accepts.

We will consider three groups of agreements. *Explicit agreements* A_{explicit} can be found by simply matching the beliefs in the initial belief stores.

$$A_{\text{explicit}} = \{ f \mid f \in I_{\text{s}} \ \&\ B_{\text{c}}(f) \in I_{\text{cs}} \}$$

In most of the cases, there are few explicit agreements. To define *implicit agreements* A_{implicit} we will consider agents' reasoners and will match the beliefs in B_{s} and B_{cs}.

$$A_{\text{implicit}} = \{ f \mid f \in \mathrm{B}_{\text{s}} \ \&\ B_{\text{c}}(f) \in \mathrm{B}_{\text{cs}} \}$$

People tend to agree when they do not contradict one another. Following an autoepistemic notion, we define *assumed agreements*:

$$A_{\text{assumed}} = \{ f \mid ((f \in \mathrm{B}_{\text{s}}) \ \&\ (B_{\text{c}}(f) \textit{ does not contradict } \mathrm{B}_{\text{cs}})) \text{ or } ((B_{\text{c}}(f) \in \mathrm{B}_{\text{cs}}) \ \&\ (f \textit{ does not contradict } \mathrm{B}_{\text{s}}))\}.$$

We will consider that a belief formula f *contradicts* a set of belief formulas F if the *negation* of f belongs to F.

3.4. Conflicts

Conflicts in interactive diagnosis are sources for negotiation and in most of the cases help in articulating learner's domain beliefs. In accordance to the definitions about agreements, we will define conflicts in computer and student's views about student's beliefs.

Explicit conflicts C_{explicit} will be obtained by matching the beliefs in agents' initial belief sets.

$$C_{\text{explicit}} = \{ f \mid ((f \in I_{\text{s}}) \ \&\ (B_{\text{c}}(f) \textit{ contradicts } I_{\text{cs}})) \text{ or } ((B_{\text{c}}(f) \in I_{\text{cs}}) \ \&\ (f \textit{ contradicts } I_{\text{s}}))\}$$

Likewise, we will define *implicit conflicts* C_{implicit} by considering all beliefs derived after agents' reasoners are applied. An ensuing definition of *assumed conflicts* would appear redundant because of its similarity with C_{implicit}.

$$C_{\text{implicit}} = \{ f \mid ((f \in \mathrm{B}_{\text{s}}) \ \&\ (B_{\text{c}}(f) \textit{ contradicts } \mathrm{B}_{\text{cs}})) \text{ or } ((B_{\text{c}}(f) \in \mathrm{B}_{\text{cs}})) \ \&\ (f \textit{ contradicts } \mathrm{B}_{\text{s}}))\}$$

3.5 Jointly constructed student model

The jointly constructed student model B (we consider only the belief part open for a discussion) consists of all agreements between the computer and the student.

$$\mathsf{B} = A_{\mathrm{explicit}} \cup A_{\mathrm{implicit}} \cup A_{\mathrm{assumed}}$$

3.6 Assigning degree of correctness to the learner's beliefs

In the previous sections we described how B - a jointly constructed learner model - will be obtained. The learner's beliefs are assigned categories of correctness. This can be done by comparing the beliefs in B with the system domain beliefs B_{c}. We consider the following categories:

$$j \text{ is a } \textit{correct belief iff } (B_s(j) \in \mathsf{B}) \;\&\; (j \in \mathsf{B}_c).$$

$$j \text{ is an } \textit{erroneous belief iff } (B_s(j) \in \mathsf{B}) \;\&\; ((\neg j \in \mathsf{B}_c) \text{ or } (j \notin \mathsf{B}_c)).$$

$$j \text{ is an } \textit{incomplete belief iff } (j \in \mathsf{B}_c) \;\&$$
$$((B_s(\neg j) \in \mathsf{B}) \text{ or } (\neg B_s(j) \in \mathsf{B}) \text{ or } (B_s(j) \notin \mathsf{B}))$$

In this section, adopting an epistemic operator belief and employing agents that do not have a deductive reasoning, we have been able to provide a mechanism for finding agreements and conflicts between a computer and a student when discuss about the student's knowledge. This allows maintaining a jointly constructed learner model. The next section illustrates the use of the mechanism in an interactive diagnostic situation.

4 The example analysed

We will now examine the example in section 2 by applying the mechanism described in section 3. The example has been elaborated extensively to show more aspects of the approach. To make the analysis simpler we will denote:

p_1 = VISUAL BASIC is an OBJECT-ORIENTED LANGUAGE.
p_2 = VISUAL C++ is an OBJECT-ORIENTED LANGUAGE.
p_3 = OBJECT-ORIENTED LANGUAGE contains OBJECTS.
p_4 = VISUAL LANGUAGES are OBJECT-ORIENTED LANGUAGES.
p_5 = VISUAL BASIC is a VISUAL LANGUAGE.
p_6 = VISUAL BASIC contains OBJECTS.
p_7 = OBJECT-ORIENTED LANGUAGE has a characteristic INHERITANCE.
p_8 = VISUAL BASIC has a characteristic INHERITANCE.

In STYLE-OLM, computer reasoners R_c are based on conceptual graphs rules of inference [17]. Some rules from R_{cs} used in the example are

$$B_c(B_s(j) \Rightarrow B_s(y)),\ B_c(B_s(j))\ ?_{Rcs1}\ B_c(B_s(y));$$
$$B_c(B_s(j) \Rightarrow B_s(y)),\ B_c(B_s(y))\ ?_{Rcs2}\ B_c(B_s(j)).$$

I_c are encoded by a domain expert in a conceptual graphs knowledge base. In the example, before the dialogue

$$I_s=\{B_s(p_1), B_s(p_2), B_s(p_3)\}$$
$$I_{cs}=\{B_c(B_s(p_1)), B_c(B_s(p_2)), B_c(B_s(p_3))\}.$$

Consulting its knowledge base, STYLE-OLM will discover that $B_c(B_s(p_1))$ is an erroneous belief – an individual has been wrongly assigned to a class. Then the system will search for possible learner's misconceptions to explain this misclassification. There are two potential candidates.

Misclassification_1. The individual has common features with an individual that belongs to the class, i.e. VISUAL BASIC is an OBJECT-ORIENTED LANGUAGE because it is a VISUAL LANGUAGE like VISUAL C++, which is an OBJECT-ORIENTED LANGUAGE.

Here, the system will start with p_2 and by *generalising* it to p_4 will *assume* that $B_c(B_s(p_2)\Rightarrow B_s(p_4))$. Then, it will make a hypothesis that $B_c(B_s(p_4))$ by applying the rule R_{cs1}.

Applying *restriction* over p_4, the system will infer p_5 and will *assume* that $B_c(B_s(p_4)\&B_s(p_5)\Rightarrow B_s(p_1))$. Then, it will make a hypothesis that $B_c(B_s(p_5))$ by applying the default rule R_{cs2}.

Misclassification_2. The individual has features that are part of the class features, i.e. VISUAL BASIC is an OBJECT-ORIENTED LANGUAGE because it contains OBJECTS.

Applying *restriction* over p_3, the system will infer p_6 and will *assume* that $B_c(B_s(p_3)\&B_s(p_6)\Rightarrow B_s(p_1))$. Then, it will make a hypothesis that $B_c(B_s(p_6))$ by applying the default rule R_{cs2}.

Thus, before the interaction the belief sets will be:

$$\mathrm{B}_s=\{B_s(p_1), B_s(p_2), B_s(p_3)\}$$
$$\mathrm{B}_{cs}=\{B_c(B_s(p_1)), B_c(B_s(p_2)), B_c(B_s(p_3)), B_c(B_s(p_4)), B_c(B_s(p_5)), B_c(B_s(p_6))\}$$

During the interaction more beliefs will be added as follows.

(4) will add $B_s(p_5)\in I_s$.

(6) will bring $\neg B_s(p_4)\in I_s$ and a challenge to $B_c(B_s(p_2))\in \mathrm{B}_{cs}$ that will imply the explicit question in (7).

(8) will show that $B_c(B_s(p_2))$ will remain in B_{cs} but $B_c(B_s(p_4))$ will be deleted. *Misclassification_1* will be withdrawn.

(10) will bring $B_s(p_6)\in I_s$.

(11) assuming $\neg B_c(B_s(p_8))\in I_c$, the computer asks about this explicitly.

(12) will show that $\neg B_s(p_8)\in I_s$.

(13) following the answer in (12), the computer will assume $B_c(\neg B_s(p_7))\in I_c$ and will aim at checking it.

(14) will confirm $\neg B_s(p_7)\in I_s$ and *misclassification_2*.

(15) will inform about the discovered misclassification.

Therefore, after the interaction

$$\mathrm{B}_s= I_s = \{B_s(p_1), B_s(p_2), B_s(p_3), \neg B_s(p_4), B_s(p_5), B_s(p_6), \neg B_s(p_7), \neg B_s(p_8)\}$$
$$I_{cs}=\{B_c(B_s(p_1)), B_c(B_s(p_2)), B_c(B_s(p_3)), B_c(\neg B_s(p_7)), \neg B_c(B_s(p_8))\}$$
$$\mathrm{B}_{cs}=\{B_c(B_s(p_1)), B_c(B_s(p_2)), B_c(B_s(p_3)), B_c(B_s(p_5)), B_c(B_s(p_6)), B_c(\neg B_s(p_7)), \neg B_c(B_s(p_8))\}$$

$$A_{\text{explicit}}=\{B_s(p_1)\ B_s(p_2)\ B_s(p_3),\ \neg B_s(p_7)\}$$
$$A_{\text{implicit}}=\{B_s(p_1)\ B_s(p_2)\ B_s(p_3),\ B_s(p_5),\ B_s(p_6),\ \neg B_s(p_7)\}$$
$$A_{\text{assumed}}=\{\neg B_s(p_4),\ \neg B_s(p_8)\}$$

$$\mathcal{B}=\{B_s(p_1)\ B_s(p_2)\ B_s(p_3),\ B_s(p_5),\ B_s(p_6),\ \neg B_s(p_7),\ \neg B_s(p_4),\ \neg B_s(p_8)\}$$

In addition, during the interaction a misclassification of type 2 was discovered as an explanation of $B_s(p_1)$.

In this example, *explicit conflicts* have not been discovered. An *implicit conflict* about $B_s(p_4)$ was discovered and consequently overcome by deleting $B_c(B_s(p_4))$ from $\mathcal{B}_{cs}$.

Before being added to the learner model, the beliefs from $\mathcal{B}$ will be assigned a level of correctness comparing them with the knowledge in the system domain model. For example, p_1 will be considered as an *erroneous* belief, p_7 will be assigned as *incomplete* and p_6 as a *correct* belief.

The example has illustrated how the mechanism described in section 3 maintains a jointly constructed learner model in an interactive diagnostic situation. Before the interaction, the learner model includes very constrained beliefs about the learner. Throughout the interaction, using its reasoners, the computer system made some hypotheses about learner's beliefs and asked him for verification. At the end of the dialogue, finding system and student's agreements, a more elaborated learner model has been obtained and a learner's misconception has been discovered.

5 Conclusion

In this paper, we have presented an approach for maintaining a jointly constructed student model in interactive diagnosis. We have adapted a belief modal operator to model the knowledge of the learner and to maintain the interaction between the computer system and the learner. The nature of interactive diagnosis where agents' reasoning is not complete and not necessarily sound has allowed us to explore several simplifications and to avoid problems due to computational complexity.

The applicability of the mechanism for maintaining a jointly constructed student model has been demonstrated in STYLE-OLM an interactive diagnosis component in a terminology learning environment.

A substantial insight from formalisation in intelligent systems is that the models developed for one application can easily be employed in another context. The mechanism for maintaining a jointly constructed learner model described above has been adjusted to natural situations of human reasoning. This brings practical advantages of the approach making feasible its extension in peer diagnosis situations where two learners discuss the knowledge of one of them. In this context, the mechanism will provide a computer system with an engine to build models of the peers as well as to mediate the interaction between them. Another potential dimension for future investigations is a possible extension of the mechanism to modelling agreements and conflicts in collaborative dialogues.

References

[1] J.A. Self, The defining characteristics of intelligent tutoring systems research: ITSs care, precisely, *International Journal of Artificial Intelligence in Education*, **10**, 350-364, (1999).

[2] J.A. Self, Bypassing the intractable problem of student modelling, in C. Frasson & G. Gauthier eds., *Intelligent Tutoring Systems: At the Crossroad of Artificial Intelligence and Education*, Ablex, NJ, 1990.

[3] S. Bull, *Collaborative Student Modelling in Foreign Language Learning*, PhD Thesis, University of Edinburgh, 1997.

[4] R. Morales, H. Pain, S. Bull & J. Kay, *Proceedings of the Workshop on Open, Interactive, and Other Overt Approaches to Learner Modelling*, Le Mans, France, July 1999.

[5] A. Kobsa, User modelling in dialogue systems: potentials and hazards, *AI & Society*, **1**, 214-240, (1990).

[6] P. Brusilovsky, Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, **6** (2/3), 87-129, (1996).

[7] V. Dimitrova, J.A. Self & P. Brna, The interactive maintenance of open learner models, in Lajoie S.P. and Vivet M., eds., *Proceedings of the 9th conference of AI in Education*, Frontiers in AI and Applications, vol. 50, IOS Press, pp. 405-412, 1999.

[8] A. Paiva and J.A. Self, TAGUS - a user and learner modelling workbench, *User Modeling and User-Adapted Interaction*, **4**, 197-226, (1995).

[9] J. Kay, The UM toolkit for cooperative user modelling, *User Modeling and User-Adapted Interaction*, **4**, 149-196, (1995).

[10] J.A. Self, *Computational Mathetics*, http://www.cbl.leeds.ac.uk/~jas/cm.html.

[11] R. Fagin, J.Y. Halpern, Y. Moses & M.Y. Vardi, *Reasoning about Knowledge*, The MIT Press, Cambridge, 1995.

[12] J. Taylor, J. Carletta, C. Mellish, Requirements for belief models in cooperative dialogue, *User Modelling and User-Adapted Interaction*, **6**, 23-68, (1996).

[13] A. Ballim and Y. Wilks, *Artificial Believers: The Ascription of Belief*, Erlbaum, Hillsdale, NJ, 1991.

[14] A. Kobsa, and W. Pohl, , The user modeling shell system BGP-MS. *User Modeling and User-Adapted Interaction* **4**(2), 59–106, (1995).

[15] S. Bull and P. Brna, Enhancing peer interaction in the Solar system, in Brna P., Baker M. and Stenning K. eds., *Roles of communicative interaction in learning to model in Mathematics and Science*, Proceedings of the C-LEMMAS conference, April 1999.

[16] V.Dimitrova, J.A. Self & P. Brna, STyLE-OLM - an interactive diagnosis tool in a terminology learning environment, in the *Proceedings of the Workshop on Open, Interactive, and Other Overt Approaches to Learner Modelling*, Le Mans, France, July 1999.

[17] J. Sowa, Conceptual graphs: draft proposed American national standard, in *Proceedings of ICCS-99 - Conceptual Structures: Standards and Practices*, Lecture Notes of Artificial Intelligence 1640, Springer Verlag, Berlin, pp. 1-65, 1999.

[18] G. Angelova, A. Nenkova, Sv. Boycheva, and T. Nikolov, CGs as a Knowledge Representation Core in a Complex Language Learning Environment, in the *Proceedings of* ICCS-2000, Darmstadt, Germany, August 2000, to appear.

[19] J. Levin & J. Moore, Dialogue games: meta-communication structures for natural language interaction, *Cognitive Science*, **1**, 395-420, (1977).