

# Helping students of introductory calculus classes: the *Leibniz* pedagogical agent

João Carlos Gluz<sup>1</sup>, Tânia Cabral<sup>2</sup>, Paulo Baggio<sup>2</sup>, Paulo Livi<sup>2</sup>

<sup>1</sup>Pós-graduação em Computação Aplicada (PIPCA)

UNISINOS – São Leopoldo – RS – Brazil

<sup>2</sup>Engenharia de Sistemas Digitais – UERGS – Guaíba – RS – Brazil

jcgluz@unisinos.br, tania.c.b.cabral@terra.com.br,  
{paulo-baggio, paulo-livi}@uergs.edu.br

**Abstract.** *This paper presents the architecture, structure of knowledge base, and the organization of the prototype of Leibniz pedagogical agent. This agent works as an educational assistant for Differential and Integral Calculus classes. Main innovative aspect present in this agent is the approach taken to model the interaction processes between students and teacher, which is based on Pedagogical Negotiation concepts. Another innovation is the inference mechanism used to detect divergences in the negotiation process. This mechanism allows probabilistic estimation of divergences in expected answers, combining a symbolic matcher of mathematical formulas with a bayesian networks inference engine. After an initial discussion about related work, the paper presents the theoretical foundations of Leibniz, shows the architecture of the agent, and the structure of its knowledge base. The organization of the prototype is also shown, including its inference process, interface facilities, and an example of use. The paper is finished with the presentation of some initial results.*

## 1 Introduction

Differential and Integral Calculus classes usually pose very hard pedagogical problems for both teachers and students. The main question that interests our research group is related to these problems, and can be stated as how to organize group-work in the classroom so that learning conditions will reach weak students. The group-work pedagogical strategies we follow to achieve these conditions are based on the perspective of *Solidarity Assimilation Group* (SAG) ideas [Baldino, 1997], supporting the notion that these strategies should aim the professional and social formation of the student.

Our research group also considers that the pedagogical and didactical use of computational tools, like *Computer Algebra Systems* (CAS), is valuable for the teaching-learning process, but do not solve all learning difficulties. These tools can even create some serious new difficulties, because to use a CAS properly is necessary that students (and teachers) expend a lot of time trying to understand not only how it works, but also eventually, how it can be programmed. These abilities can be interesting for advanced students, but surely, they do not help very much a novice trying to understand the basic concepts of Calculus.

Considering this our group has designed and developed the *E-M@T* web-based learning environment [Cabral, 2005], which incorporates SAG concepts and allows the use of CAS systems in an intuitive and natural way. *E-M@T* was developed to give support for teaching and learning activities in Calculus classes for engineering courses. Recently it was evolved to support other kinds of Engineering classes like Electrical Circuits, Analytical Geometry and Linear Algebra. The system is in experimental use since 2005 in Calculus classes in the Digital Systems Engineering Course.

The development and experimentation of *E-M@T* has led us to seek ways to make the interaction and interface between *E-M@T* system and the student not only natural and intuitive, but also more appealing, challenging and intelligent. The possibility to use ideas and technologies from *Intelligent Tutoring Systems* (ITS) [Self, 1998] and *Intelligent Learning Environments* (ILE) [Fernandes-Manjon *et al.*, 1998] was considered a natural way to incorporate these features to *E-M@T*. However, one important problem

we found in this new (for us) research line was the fact that there are relatively few works on pedagogical agents specifically designed for Calculus domain.

The ITS research line that best fit our purposes was based on negotiation mechanisms that can be used in learning environments. Discussions about the use of negotiation mechanisms in learning environments are not recent. According to [Self, 1992], there are two major motivations for the use of negotiation in ITS: i) they make possible to foster discussions about how to proceed, which strategy to follow, which example to look for, in an attempt to decrease the control that is typical of ITS, and ii) they give room for discussions that yield different viewpoints (different beliefs), provided that the agent (tutor) is not infallible. The notion of some form of negotiation occurring in the classroom is not only compatible with SAG principles and practices, but, it is clearly complementary to SAG methods. The operational use of contracts of work in the classroom, linked to rewards based on measured work, imply in some form of negotiation to be realized before, during and after contracts be established and measurement instruments be defined and used. Surely, this negotiation must follow a pedagogical rationale, that is, it must be a form of *Pedagogical Negotiation* (PN) ideas [Flores *et al.*, 2005; Gluz *et al.*, 2006; Seixas *et al.*, 2006].

This reduced the research problem studied in the present work, on how to design a pedagogical agent [Gürer, 1998] that can incorporate negotiation strategies, which are specific to Calculus domain, and how this agent can be properly integrated in the structure of E-M@T system, and handle its learning objects. The main purpose of this agent, called *Leibniz* is to bring Artificial Intelligence techniques to E-M@T interaction and interface facilities, serving as an assistant in the educational processes that occur in Differential and Integral Calculus classes.

The application domain of *Leibniz* is focused on differential and anti-differential calculus of elementary functions. The student model of *Leibniz* is a probabilistic version of the perturbation model [Beck *et al.*, 1996] based on Bayesian Networks [Pearl, 1993; Cowel *et al.*, 1999]. Due to PN analysis, it was possible to limit the application of this model to the estimation misconceptions and divergences that students have in respect to some particular exercise of E-M@T.

Another important motivation for the present research is to use *Leibniz* for interoperability experiments with other learning environments and as an analysis tool for the identification and reuse of domain independent pedagogical knowledge. In particular, there is a great interest to setup interoperability experiments between *Leibniz* and ILEs that already follow PN concepts, trying to reuse and apply general negotiation strategies in the context of Calculus learning domain.

In the next sections, these ideas will be delineated. Section 2 will present works about learning environments and discuss how they are related to *Leibniz*. Section 3 will present theoretical foundations that guide the research, design and development of *Leibniz*. The following section will discuss architectural aspects of the agent. Section 5 will present the characteristics of the knowledge base of *Leibniz*. Section 6 will describe the organization of the prototype, including inference process, interface facilities, and example of use. Finally, Section 7 will present the initial results obtained with the application of the prototype and the status of development of *Leibniz*.

## 2 Related Work

Although there exists an extensive literature on ITS and ILE [Beck *et al.*, 1996] and even knowing that Calculus is an interesting and important learning domain, we found relatively few works that can be applied directly to our research. There are few examples of ITS or ILE systems that work with domains similar to Calculus, and use probabilistic representation mechanisms and pedagogical diagnostic techniques related to our research. *DT Tutor* system [Murray *et al.*, 2001], which proposes a decision-theoretic architecture for agents intended to select tutorial discourse actions, is the ITS with more points in common with our work. This system was applied to several learning domains, including calculus problem-solving (the *CTDT* selection engine), and uses Dynamic Decision Networks (a variant of bayesian networks that support decision nodes) to represent tutor strategies, and student models. Also worth of note are *ANDES* [Conati *et al.*, 2002], which is focused on the learning of basic Newtonian Physics, and *ACE* [Bunt and Conati, 2003], which supports the learning of basic operations with functions. These systems work with learning domains that present similarities to Calculus, and both use probabilistic student models, represented by bayesian networks, to guide their tutoring activities.

The *AMPLIA* learning environment [Flores *et al.*, 2004; Vicari *et al.*, 2003], which works with the medical diagnosis domain, is also important to our work because it uses PN ideas [Flores *et al.*, 2005; Gluz *et al.*, 2006; Seixas *et al.*, 2006] to analyze and model student-system interaction. Despite the very different learning domains of these systems, *AMPLIA* and *Leibniz* share the same theoretical basis for the analysis and modeling of interaction phenomena that occur in the classroom. In being so, there is a good chance that they can interoperate, if standard agent communication and content languages [Gluz and Vicari, 2003] are used, and if an appropriate ontology is created to specify shared concepts.

Different from *DT Tutor*, *ANDES* and *ACE*, *Leibniz* do not intent to model the entire learning process that students should develop in relation to each study topic, but concentrate its efforts only on specific interactions that can occur in a particular topic of study. Interactions possible to occur in a study topic are defined by the corresponding learning object (called *FTE*) in the E-M@T system, through lists of exercises and experiments contained in the object. FTEs are objects created by teachers that define the overall pedagogical structure and corresponding didactics to be applied with the learning topic. Epistemology and didactics used in each FTE are based on the Baldino's works that recovers the didactical use of classical epistemology for calculus conceptions, which support the notion of infinitesimals, now rigorously founded on Hiperreal Theory of Robinson [Baldino, 1995; Baldino, 1998]. The pedagogical and didactics strategies incorporated in each FTE are constructed under the perspective of SAG ideas [Baldino, 1997].

*Leibniz* will participate in E-M@T interaction with students as a pedagogical agent trying to help students in their learning process. PN analysis assumes that this interaction can be considered an open-ended non-economical form of negotiation intended to solve judgment differences and conflicts that occur in the classroom only by pedagogical means. The PN approach to analyze teaching-learning processes is that these processes should not be necessarily considered guided processes, but considered continuous discussion processes where the teacher needs to use pedagogical strategies and tactics to help students to advance their learning in a productive way. In being so, one difficult initial problem of PN analysis is how to assess where the negotiation process is at any moment of time, that is, how the teacher will evaluate student's current work. Only when this assessment is available, it is possible to select new actions to advance the teaching-learning process.

In the case of *Leibniz*, this assessment depends on how to discover what are the misconceptions or reasoning divergences the student is having in relation to expected answers to FTE's exercises and experiments. To make appropriate assessments about these misconceptions and errors, diagnostic rules used in *Leibniz* were defined not only on our experience as teachers, but also borrow some ideas and techniques from works about error analysis in Mathematical Education [Cury, 2004] and in Differential and Integral Calculus [Mariani, 2005].

The *MAXIMA* CAS [MAXIMA, 2007], which is an open-source branch of old MIT's *MACSYMA* system, is important to our work because it is the backend CAS used in *E-M@T*. Answers to proposed exercises must be returned as formulas understood by *MAXIMA*. These formulas are the basic symbolic structure used in the interaction with students. Therefore, to assess how the teaching-learning process is evolving, the *Leibniz* agent should understand the structure of these formulas, and be able to compare them with expected results from the exercise.

### 3 Pedagogical Negotiation

The first step to understand what we consider *pedagogical negotiation* is to know what can be the role of negotiation within a teaching-learning process. To do this we need to understand clearly, which the final objectives of this process are and how negotiation may help to reach them. The problem here is how achieve these goals without loosing the pedagogical aspects of this process. For example, if we try to use traditional negotiation concepts, based on the Economic Theory, the expect result of some negotiation is the maximization of participant's gains. Negotiation here is essentially a search process that looks for a solution, which will maximize the gains of agents in relation to all other possible solutions. Gains are measured through a *utility function* known by the agents. The problem lies exactly on the presupposition that an agent knows how to determine the utility in a given situation, as well as in situations derived from its actions [Sandholm, 1999]. This does not happen in a teaching-learning process, because, it is difficult to realize how a student generates all his/her preferences.

The same is valid for the teacher. Simply, it is not reasonable to presuppose that the teacher has total knowledge on all situations that may happen in a teaching-learning process. Students can present results so that, even they are not in accordance with the teacher's expectations, they can be perfectly acceptable in terms of teaching objectives intended.

We observe that results of a pedagogical negotiation should be related to the final objectives of teaching-learning process, as well as the concept of preference or utility for an agent are not enough to characterize results expected in the pedagogical negotiation. As a solution for these problems, we adopted some simplifying presuppositions, based on the common sense, with which we expect to contribute to elicit more this issue.

A primarily presupposition is not to approach the teaching-learning process directly as a *knowledge transference* process. Characterizing the process in this way implies to consider the need for solving classical epistemological issues that do not have concrete answers: what is knowledge? How could it be transferred to another person? How to measure a person's knowledge? To this claim, we add the discourse of pedagogical models supported in the Piaget's genetic epistemology [Piaget, 1970], where the subject builds knowledge through interactions.

We will consider the notion of confidence that an agent would have in relation to another agent (or about itself), aiming at maximizing this relationship as the process evolves. We will adopt the notion of confidence based on the expectation of future behavior of an agent in relation to another (or to itself). The idea is that the "expectation of future behavior" may be evaluated more precisely than the perception of "how much this agent knows on theme".

Considering the teacher-student learning scenario, a first step in the characterization of the teaching-learning process is to attribute distinct objectives for each role. Relating to the teacher's role, as a mediator of the teaching-learning process, it should be considered not only the relationship of confidence between teacher and student, but it is required an inverse analysis, that is, the relationship of confidence between student and teacher. Thus, there are two important characteristics to outline in the student's behavior. (1) The student is confident on the teacher's appraisal capacities during the development of contents. This statement does not imply necessarily that the confidence level is complete, i.e., that the student should blindly trust the teacher. What is said is that there should be a reasonable level of confidence, and that it should be undertaken so that the teaching-learning process can be accomplished. (2) Definition of what the student expects because of the teaching-learning process. The simplifying presupposition is to undertake that the student expects to reach a level of knowledge that makes possible to understand and solve situations or problems within the area or discipline that is being studied. The point is not exactly what the student intends, but how we could have concrete evidences that this objective was reached.

The teaching-learning process could be seen as a way of reducing the initial asymmetry of the confidence relation between teacher and student, in respect to the topic studied, maximizing the confidence of all. Putting this into a scheme:

- Initial state of the pedagogical negotiation process:
  - Teacher: **(IP.1)** High level of confidence in his own knowledge.  
**(IP.2)** Low level of confidence in the student's knowledge.
  - Student: **(IA.1)** Low level of confidence in his own knowledge.  
**(IA.2)** High level of confidence in the teacher's knowledge.
- Final (expected) state of the pedagogical negotiation process:
  - Teacher: **(FP.1)** High level of confidence in his own knowledge.  
**(FP.2)** High level of confidence in the student's knowledge.
  - Student: **(FA.1)** High level of confidence in his own knowledge.  
**(FA.2)** High level of confidence in the teacher's knowledge.

Conditions **(IP.1)** and **(FP.1)**, as well as **(IA.2)** and **(FA.2)** should not change, being only bases for an adequate beginning, development and end of the process. The effective result of the process would be to increase the confidence level of the teacher in the student, from **(IP.2)** to **(FP.2)**, and of the student in himself/herself, from **(IA.1)** to **(FA.1)**.

Although the teaching-learning process can be considered a process of "equalizing" confidence levels, we are not presupposing that it has a linear, monotonic and continually increasing behavior. There is no

guarantee that this process will increase a certain confidence level step-by-step (linear), without interruptions (monotonic) or drawbacks (continually increasing) until it reaches the required level. Evaluation conflicts between teacher and student are common, and so it is needed to adopt a negotiation process in order to solve them through argumentation mechanisms, with the aim of strengthening the confidence relation between teacher and student. The expected result will be to increase the confidence level among all agents.

Another important issue is the definition of with teaching and learning strategy should be selected. The present work undertakes the approach introduced in [Seixas *et al.*, 2006], where the student will have an active role in the learning process and the teacher will be the mediator and motivator of this process proposing reflection strategies in problem solving. The main pedagogical strategies adopted are positive strategies turned towards student's motivation, and not only negative ones to identify mistakes and problems of students.

Considering these discussions, the computational model we are using to represent pedagogical negotiation assumes that teacher role is performed by three software agents: *Learner*, *Domain*, and *Mediator* agents. *Learner* agent, in addition to representing the student, also infers the self-confidence and credibility level, as a teacher that observes the students' actions. *Domain* agent evaluates the quality of students' solutions in respect to the learning domain. *Mediator* is the agent responsible by the selection of pedagogic strategies and for the successful conclusion of pedagogical negotiation.

The approach of pedagogical negotiation can be applied to areas of knowledge that share some characteristics such as incomplete knowledge and different points of view or even domains where there is no "knowledge" – considered in its classical definition, in which knowledge is always something true – but a set of justified beliefs about what one can argue and debate. These characteristics foresee the transformation of viewpoints, both from the system and from student, into beliefs instead of knowledge. This implies a special type of teaching dialogue, if an interactive change of justified beliefs is a simplified definition of argumentation [Schwarz *et al.*, 2001]. This is a complex process because it involves the student's autonomy, the symmetry of relations between teacher and students or among agents, and the levels of flexibility, which involve the agents' level of freedom to perform their actions [Baker, 1994]. This work does not see the presence of 'conflict' (either openly declared or acknowledged or not) as essential in the definition of negotiation. The basic requirement is that the interaction among agents shares a common goal so that an agreement is reached with respect to the negotiation object. Usually, different dimensions of the negotiation object will be negotiated simultaneously. The initial state for a negotiation to take place is the absence of an agreement, which can include a conflict or not.

## 4 Architecture of Leibniz

The architecture of *Leibniz* agent is shown on Figure 1. From the point of view of *E-M@T*, *Leibniz* runs as a backend subsystem. The main point of interaction with *E-M@T* system is provided by *SASB-Leibniz* component of *E-M@T* (SASB is the generic name for backend interface subsystems of *E-M@T*). *Select Diagnostics*, *Select Recommendation* and *Estimate Self-confidence and Credibility* are the active components of *Leibniz*. Currently they work as functional components working through client-server interactions, which are responsible by tasks related to PN process. Eventually they will become autonomous sub-agents of *Leibniz*.

From the point of view of *Leibniz*, the *E-M@T* system provides the interface facilities used to communicate with users (students). Interactions are started when the student propose to *E-M@T* (with *Leibniz* behind) a particular solution for some test or question proposed in the FTE. After that, *Leibniz*, as an assistant agent, will use the student's solution as an evidence of where the learning process is currently "positioned", and then, based in the current and previous states of this process, *Leibniz* will demand to select actions to be taken to advance this process. The first task of *Leibniz* is proceed with the *pedagogical diagnostic* of the student learning status (*Select Diagnostic* component), and the second task is select *mediation strategies and tactics* for pedagogical purposes that will result in recommendations or advises to be sent to student (*Select Recommendation* component).

In terms of PN process, the *Select Diagnostic* component forms the core of the Domain specialist agent, being responsible for representing the expert in the domain. Pedagogical diagnostic tasks in *Leibniz* are based on pedagogical *Diagnostic Rules*, that are sets of rules associated to each test and exercise of the FTE used to diagnostic possible errors in the problem solving student's reasoning to solve the test or exercise proposed by the FTE. Diagnostics are always approximate, subjective estimations of which are the most probable problems in the student learning process. To support this probabilistic estimation diagnostic, *Leibniz* represents *Diagnostic Rules* through bayesian networks, which are the most commonly used representation formalism and inference engine for probabilistic knowledge.

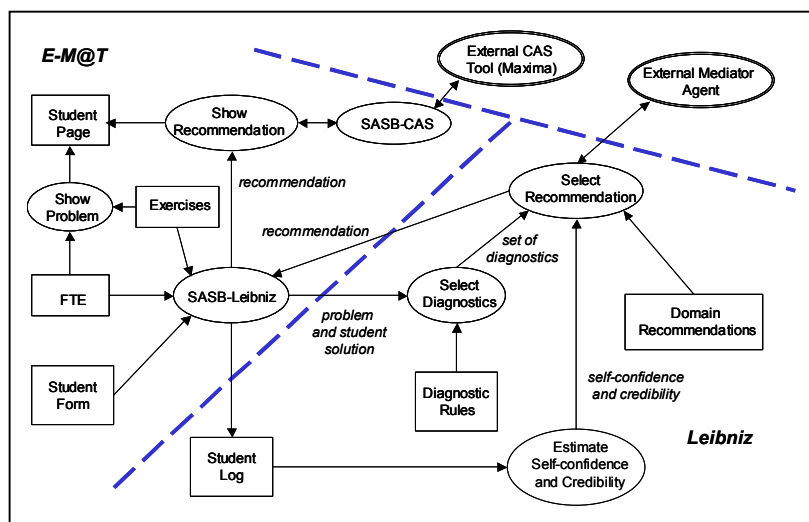


Figure 1: Leibniz architecture

The *Select Recommendation* component of *Leibniz* performs some roles of the *Mediator* agent, being responsible for mediating the teaching-learning process between the *Domain* agent and the student. The *Select Recommendation* implementation of this role is limited, in the sense that this component only had information about how to map pedagogical strategies and teaching tactics to the learning domain, and knowledge on how to interact with an external *Mediator* agent that will identify strategies and tactics appropriate for the situation. Indeed one of the main goals of the project is to reuse the mediation knowledge already incorporated by AMPLIA's *Mediator* agent [Flores *et al.*, 2005]. This agent already communicates with other agents through a standard agent communication language (the FIPA-ACL language), thus *Select Recommendation* component will interact with it through this language, informing the current state of the teaching-learning process and receiving, in return, strategies and tactics to be applied to the current situation. The *Domain Recommendations* knowledge base contains a set of rules that maps these strategies and tactics to the learning domain of *Leibniz*.

The *Estimate Self-confidence and Credibility* component also implements a limited role of the *Learner Specialist* agent of pedagogical negotiation. The basic goal of this component is to infer – such as a teacher would do – the levels of grasp of consciousness of the student, after the observation of student's actions that are stored in the *Student Log* database. This inference will result in the self-confidence and credibility levels (variables) used by AMPLIA's *Mediator* agent.

## 5 Knowledge Base Structure

The *Leibniz* Knowledge Base (KB) is organized by the exercises available in the *E-M@T* system. For each exercise, there exist one specific set of pedagogical diagnostic rules aiming the identification of misconceptions and reasoning errors.

Observed divergences are used as evidences for which kind of reasoning error or misconception the student should be doing. In *Leibniz*, these divergences are detected by the comparison of the formula given by the student as the solution of the problem, with the formula template used to trigger the rule. If a matching occurs then the rule is considered for diagnostic purposes. However, it is important to note that the matching does not activate the rule in the classical sense of production systems, the matching must be

considered only as evidence to be taken in a Bayesian network inference process. The resulting diagnostic will be only a probabilistic estimation of the situation, based on conditional probabilities given the evidences.

Errors and misconceptions are identified by these rules through the association between an error name and a confidence degree attributed by the system to the fact that found evidences (divergence in relation to expected solution) implies the diagnostic. The confidence degree is the subjective probability attributed by the system that the diagnostic is correct. Confidence degrees are expressed by *weights*, ranging from zero to 10, attributed to diagnostics. Associated to each type of misconception or error there is a set of mediation rules that provide pedagogical recommendations associated to the error. Recommendations selected are sent to the student considering pedagogical strategies and tactics that should be appropriate to the student.

## 5.1 Diagnostic Rules

Diagnostic rules are defined through Prolog clauses  $diag(DT, FT, WT)$ , where  $DT$  is a literal symbol which identifies the type of pedagogical diagnostic type,  $FT$  is a formula template represented by a Prolog term and  $WT$  is the weight of the diagnostic, expressed by an integer numeric value between 0 and 10.

$FT$  terms are templates that represent mathematics formulas expected by the rule. These templates must be compared with the student's answer and, if the answer and  $FT$  term match so the rule will apply. This matching process is not implemented by simple Prolog unification. We developed for *Leibniz* agent a special symbolic matcher for mathematical formulas, which allows parametric templates, supports sub-formula meta-variables, recognizes standard commutability and associability of summing and multiplication operators and implements reduction of formulas composed exclusively of literal terms.

Figure 2 presents a subset of the pedagogical diagnostic rules used for the exercise that requests the resolution of the differential of formula ' $\sin^2(x)$ '. This exercise checks the ability of the student to apply the chain rule and differentiation rules for powers and trigonometric functions to solve this kind of problem. To solve it, it is necessary to apply the chain rule, then differentiation rules for powers and for ' $\sin(x)$ ' function, resulting in ' $2 \sin(x) \cos(x) dx$ '. This set of rules uses three parameters, to allow that other similar exercises reuse the same structure.

```

% Exercise: d(sin(x)^2)
% Parameters
param(c, 2).
param(f1, sin({x})).
param(f2, cos({x})).
param(dx, d({x})).

% Diagnostic Clauses
diag(correct,      c*f1*f2*dx,      10).
diag(partial,     c*f2*f1,          10).
diag(chain_rule,  c*{x}*f2*dx,      10).
diag(chain_rule,  c*{x}*f1*dx,      9).
diag(chain_rule,  {}*f1*f2*dx,      8).
diag(sin_rule,    -c*f1*f2*dx,      9).
diag(power_rule,  c*{x}*f1*f2*dx,    8).
diag(power_rule,  {x}*f1*f2*dx,     9).
diag(interp,      {}*cos({x}^2)*dx, 9).
diag(interp,      {}*f2,            8).

```

Figure 2: Example set of pedagogical diagnostic rule

Diagnostic clauses in this exercise check for possible errors in the application of chain rule (the *chain\_rule* type of diagnostic) of differentiation, errors in the application of differentiation rules for powers (*power\_rule* diagnostic) or for trigonometric functions (*sin\_rule* diagnostic). They also verify if the answer is partially correct (*partial* diagnostic) or if the student had misinterpreted the exercise's goals (*interp* diagnostic).

The subset of  $diag(DT, FT, WT)$  clauses in a diagnostic rule, should be understood as a symbolic compact form for the correspondent Bayesian network to be used in the diagnostic reasoning. In this way, the subset of  $diag(DT, FT, WT)$  clauses presented in Figure 2 generates the BN shown in Figure 3. This BN defines the qualitative model of the probabilistic inference process used to do the pedagogical diagnostic.





(5.1)  $p(c*f1*f2*dx)=0.5$   
 $p(c*f1*f2)=0.5$   
 $p(c*\{x\}*f2*dx)=0.5$   
 $p(...)=0.5$

(5.2)  $p(correct | c*f1*f2*dx)=1.0$   
 $p(correct | \neg c*f1*f2*dx)=0.0$   
 $p(sin\_rule | -c*f1*f2*dx) = 0.9$   
 $p(sin\_rule | \neg \neg c*f1*f2*dx) = 0.1$   
 $p(partial | c*f1*f2)=1.0$   
 $p(partial | \neg c*f1*f2)=0.0$   
 $p(chain\_rule | c*\{x\}*f2*dx, c*\{x\}*f1*dx, \{\} *f1*f2*dx)=1.0$   
 $p(chain\_rule | c*\{x\}*f2*dx, c*\{x\}*f1*dx, \neg \{\} *f1*f2*dx)=1.0$   
 $p(chain\_rule | c*\{x\}*f2*dx, \neg c*\{x\}*f1*dx, \{\} *f1*f2*dx)=1.0$   
 $p(chain\_rule | c*\{x\}*f2*dx, \neg c*\{x\}*f1*dx, \neg \{\} *f1*f2*dx)=1.0$   
 $p(chain\_rule | \neg c*\{x\}*f2*dx, c*\{x\}*f1*dx, \{\} *f1*f2*dx)=0.9$   
 $p(chain\_rule | \neg c*\{x\}*f2*dx, c*\{x\}*f1*dx, \neg \{\} *f1*f2*dx)=0.9$   
 $p(chain\_rule | \neg c*\{x\}*f2*dx, \neg c*\{x\}*f1*dx, \{\} *f1*f2*dx)=0.8$   
 $p(chain\_rule | \neg c*\{x\}*f2*dx, \neg c*\{x\}*f1*dx, \neg \{\} *f1*f2*dx)=0.2$

$p(power\_rule | \{x\}*f1*f2*dx, c*\{x\}*f1*f2*dx) = 0.9$   
 $p(power\_rule | ...) = ...$

$p(interp | \{\} *f2, \{\} *cos(\{x\}^2)*dx) = 0.9$   
 $p(interp | ...) = ...$

(5.3)  $p(divergence | correct) = 0.0$   
 $p(divergence | \neg correct, partial) = 1.0$   
 $p(divergence | \neg correct, chain\_rule) = 1.0$   
 $p(divergence | \neg correct, sin\_rule) = 1.0$   
 $p(divergence | \neg correct, power\_rule) = 1.0$   
 $p(divergence | \neg correct, interp) = 1.0$   
 $p(divergence | \neg correct, \neg partial, \neg chain\_rule, \neg sin\_rule, \neg power\_rule, \neg interp) = 0.5$

Figure 4: Quantitative model of the BN presented in Figure 3

## 5.2 Domain Recommendations

Domain recommendations link pedagogical strategies and teaching tactics that can be used for some particular type of diagnostic with the learning domain of *Leibniz* (Calculus). They are specified by a independent base of  $recomm(DT,ST,TT,RE)$  clauses, where  $DT$  is a Prolog term that identifies the diagnostic,  $ST$  is the identifier of the pedagogical strategy associated to the recommendation,  $TT$  the identifier of teaching tactic and  $RE$  is a text expressing the recommendation associated to the diagnostic. If  $ST=all$  then the recommendation is applied to all strategies. In the same way if  $TT=gen$  then tactics implemented by the recommendation is generic.

*Leibniz* will use the same classification of pedagogical strategies and teaching tactics adopted in AMPLIA system [Seixas *et al.*, 2006], which are understood by the Mediator Specialist agent used in this system. The following classes of strategies will be considered for pedagogical purposes:

- *Orientation*: The goal is to open new spaces for the student in case he builds a network that is not feasible. Direct information is provided to the student so that he can build the network in a different manner (so that his network becomes a BN).
- *Support*: This strategy also presents concrete and contextualized data, so that the student can increase his confidence;
- *Contest*: This strategy aims at warning the user about inconsistencies in his network, fostering a new assessment, so that the student can redo some procedures based on those that presented good results. The procedures of conservation and decomposition are involved in the Contest strategy.
- *Confirmation* and *Widening*: these strategies are directed towards the third level of grasp of consciousness, requiring reflected abstractions, such as the variation of experimentation factors and construction of new hypothesis. Confirmation takes place through the presentation of data and hypotheses, which aim at making the student, reflect and increase his self-confidence, and Widening aims at stimulating the production of new hypotheses.

A strategy is understood as a plan (cognitive process) that aims at reaching an objective, but that must be accomplished, performed or, in this case, presented to the student through a teaching tactics. Thus, to each strategy class previously elaborated was added basic tactics, depending on different levels of student's autonomy:

- Orientation strategy: tactics are Correction (to redo the action), Indication (to follow some indication) and Suggestion (to decide, based on concrete material).
- Contest strategy: tactics are Experimentation (possibility of manipulating data), Search (search data) and Reflection (analyze data);
- Support strategy: uses the Example tactics, which is "to do the same";
- Validation strategy: tactics are: Demonstration, which means the observation of data presented, and the expert's model Presentation;
- Widening strategy: tactics are Discussion (using an argumentative text), Hypothesis (formulation a hypothesis to anticipate an action) and Problem Proposal, which is the data organization.

Figure 3 presents a small set of possible domain recommendations. These clauses map the *correct* diagnostic in all strategies and tactics to a single congratulation phrase. The *partial*, *power\_rule*, *sin\_rule*, and *interp* diagnostics are also mapped to specific single advices in all strategies and tactics. In the case of *chain\_rule* diagnostic distinct recommendation are passed to the student depending on the strategy or tactics used. The clause with *all* strategies and *generic* tactics works as a default rule, which is applied only if there is no match with other specific strategy, and tactics. Specific strategies understood by the Mediator Specialist agent are identified by their names: *orientation*, *contest*, *support*, *validate* and *widening*. Valid teaching tactics are identified by their names: *correction*, *indication*, *suggestion*, *experimentation*, *search*, *reflection*, *example*, *demonstration*, *presentation*, *discussion*, *hypothesis* and *problem\_proposal*.

```
% Domain Recommendations
recomm(correct,all,generic,'congratulations, correct answer.').
recomm(partial,all,generic,'almost OK, only a small detail.').
recomm(chain_rule,all,generic,'review the chain rule').
recomm(chain_rule,contest,reflection,'how chain rule is applied?').
recomm(chain_rule,orientation,correction,'please, remake previous exercise').
recomm(chain_rule,orientation,suggestion,'review the chain rule, see FT20').
recomm(power_rule,all,generic,'take a look at powers rule.').
recomm(sin_rule,all,generic,'please, review differentiation of sin()').
recomm(interp,all,generic,'read carefully the problem statement').
```

**Figure 5:** Example set of recommendations

## 6 Leibniz Prototype

The development of the prototype and its successful integration with EM@T showed us that the architecture proposed for *Leibniz* is sound and that the general approach of design and development is appropriate. Currently, the prototype of *Leibniz* already passed the first integration tests with *E-M@T* and started to be used in real experiments in the classroom.

The prototype of *Leibniz* was implemented in SWI-Prolog 5.0. Currently the knowledge base of the prototype is composed by 25 sets of pedagogical diagnostic rules, covering differentiation exercises for polynomial, trigonometric, exponential and logarithmic functions used in the initial FTEs of Calculus classes. The prototype is been developed in evolutionary and incremental form, being continuously evaluated with the results of its use in classroom and incorporating improvements based in these evaluations.

The knowledge base of *Leibniz* can be edited or reviewed by designers and teachers through a web editor included in *E-M@T*. The editor already allows a limited form of editing sets of diagnostic rules and the association of the set to the correspondent exercise. The prototype is operational since the beginning of 2006. The prototype also includes an evaluation tool that asks students, at the end of the class, their opinion about the effect of *Leibniz* in the problem solving process.

Main goal of first version of the prototype was to provide basic knowledge representation format and tools that are able to handle mathematical formulas and diagnostic rules and check the viability of the overall approach. The work was focused on the *Select Diagnostic* component. The component responsible for select the pedagogical strategies and tactics (*Select Recommendation*) was implemented in a limited way. The component that identifies the self-confidence and credibility of the student, and the interaction with external agents were not implemented in the first version. However, it was already defined the knowledge representation format that will be used to map pedagogical strategies and tactics to the Calculus domain (the *Domain Recommendation* base of clauses).

Leibniz interfaces with *E-M@T* system through *SASB-Leibniz* module. Through a simplified user interface, shown in Figure 6, the student selects one of the options available for him in the *SASB* form, fill the text fields with your partial answer and send a requisition for *SASB*.

This subsystem receives the requisition, simplifies the partial answer, and verifies which next steps need to be taken to complete the user requisition. When *Leibniz* help is available for the exercise, then *SASB* will activate the *Leibniz* agent supplying it with student answer together with the diagnostic KB associated to that exercise.

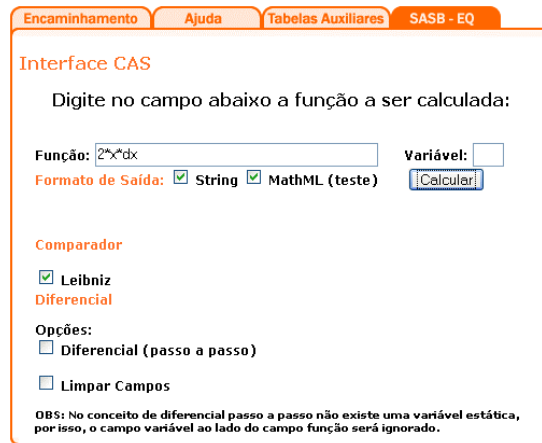


Figure 6: Leibniz user interface in E-M@T system

The set of diagnostic rules presented in Figure 2, correspond to an exercise that requires the student to find  $dy$  when  $y = \sin^2(x)$ . The answer is obtained first applying the chain rule, and then differentiation rules for powers and for ' $\sin(x)$ '. The correct answer is  $dy = 2\sin(x) \cos(x) dx$ .

The interaction starts when the first answer is submitted. Supposing that the first answer given by the student is ' $x \cos(x^2) dx$ '. This solution must be encoded in a text expression recognized by the system:

$$x * \cos(x^2) * d(x) \quad (5.1)$$

This expression matches only with diagnostic clauses  $diag(interp, \{ \cos(x^2) * dx, 9 \})$  (see Figure 2), resulting in the misinterpretation diagnostic (*interp*) having a high probability (weight) of 0.9. According to the recommendation clauses presented in Figure 5, the correspondent recommendation sent to student is "read carefully the problem statement".

Next student's attempt is represented by the following text expression:

$$2 * x * \cos(x) * d(x) \quad (5.2)$$

The expression (5.2) matches with clause  $diag(chain\_rule, c * \{x\} * f1 * dx, 9)$  giving a high probability (0.9) to the possibility that the student had misapplied the chain differentiation rule. Currently *Leibniz* prototype works only in orientation strategy mode, sending suggestions on how the student can proceed. Thus *Leibniz* send the recommendation "revise the chain rule, see FT20".

After reviewing how to use the chain rule the student makes a new attempt considering this use. The next answer is:

$$2 * x * \sin(x) * \cos(x) * d(x) \quad (5.3)$$

The expression (5.3) matches with two clauses:  $diag(chain\_rule, \{f_1 * f_2 * dx, \delta\})$  and  $diag(power\_rule, c * \{x\} * f_1 * f_2 * dx, \delta)$  (see Figure 2), giving equal probability to the possibility that there is some problem in the application of the power differentiation rule or in the chain differentiation rule. Following its inference algorithm, *Leibniz* prototype sends both recommendations to the student.

After checking the application of power rule the student send the correct answer:

$$2 * \sin(x) * \cos(x) * d(x) \quad (5.4)$$

## 7 First Results

Several monitored experiments were conducted with *Leibniz* prototype in the first semester of 2006 with introductory Calculus classes of the Digital Systems Engineering course of UERGS. The main purpose of these experiments was to validate the user and *E-M@T* interface of the agent, evaluate and adjust first version of agent's KB and check the initial impact of the agent in learning process.

Some interesting preliminary qualitative results of pedagogical nature were also observed in these initial experiments. We detected, for example, that some important conceptual errors were not considered in the first versions of diagnostic rules, which were focused on detecting errors in the application of derivation rules like the chain rule. A conceptual error not related to application of derivation rules was detected after the observation of agent's interaction log and confirmed with interviews and extra rounds of exercises with the students. The problem was that students had serious difficulties to understand the concept that '*d*' operator, as the differential of a variable, is an operator over functions (is a functional operator) and not another function. Consequently, we need to extend diagnostic rules to detect this kind of error based on appropriate evidences observed in the student's formulas.

Another interesting fact was the observation of an unexpected learning tactic employed by students to "guide" the behavior of agent: students with learning troubles explored an "escape door" from *Leibniz*, systematically sending answers entirely unexpected by the agent. When this occurs the agent returns a message that it cannot help and frees the student to use *MAXIMA* CAS to obtain the correct answer from the CAS. On the other hand, students with better abilities started to produce answers with clear intention to test the limits of the agent. These observations implied in the necessity to adjust and extend the rules to handle these possibilities, not foreseen in the initial design of the rules.

### 5.4 Future Developments

Currently the *Leibniz* prototype does not support Bayesian inference in diagnostic rules. It uses a simplified inference method based on weight maximization. In being so, the main development goal of *Leibniz* project in 2007 is to incorporate to the agent a Bayesian inference engine and a translation mechanism from symbolic representation of diagnostic rules to Bayesian Networks. The Bayesian inference engine used in *AMPLIA* will also be used in *Leibniz*. To this purpose, the engine is being isolated from the rest of the system as a stand-alone application to be interfaced with SWI-Prolog.

Another important step will be the start of interoperability tests with *AMPLIA* planned for the end of 2007. Presently, *Leibniz* uses a simple method to select the recommendation, picking the first it encounters and not considering teaching strategies and tactics informed in the recommendations. The idea here is to start the interoperability tests with *Mediation* agent from *AMPLIA* allowing the reutilization of the pedagogical mediation knowledge embodied in this agent. To this purpose is being planned the inclusion of FACIL library [Gluz, 2002] to *Leibniz*, that will allow the use of FIPA-ACL [Gluz and Viccari, 2003] to implement the communication between agents.

A long-term goal of the project is to specify how to estimate students' self-confidence and credibility based on their logged activity. This goal is currently in the empirical phase. We are now collecting student logs to have data for the analysis phase. Another long-term goal is to create and incorporate inference processes that will be able to automatically generate the diagnostic rules associated to a new exercise. The first step to achieve this goal is to accumulate a considerable knowledge base of well-tested diagnostic rules. This implies a lot of design work and experimentation with the prototype of *Leibniz*. This work will be done during all 2007, by the use of the agent in the classroom with the corresponding testing, validation, refinement and extension of the agent's KB.

**Acknowledgments.** The authors gratefully acknowledge the Brazilian agencies FAPERGS and CNPq for the partial support to this research project.

## References

- BAKER, M.J. 1994. A model for negotiation in teaching-learning dialogues. *Journal of Artificial Intelligence in Education*, 5 (2): 199-254.
- BALDINO, R. R. 1995. Cálculo infinitesimal: passado ou futuro? *Temas & Debates*. Sociedade Brasileira de Educação Matemática, 6: 5-21.
- BALDINO, R. R. 1997. Student Strategies in Solidarity Assimilation Groups. In Zack, V. Mousley, J. Breen, C. (Eds.) *Developing Practice: Teacher's inquiry and educational change* (pp. 123-134). Geelong, Australia: Deakin University.
- BALDINO, R. R. 1998. *Desenvolvimento de Essências de Cálculo Infinitesimal*. Rio de Janeiro: MEM/US
- BALDINO, R. R. and SILVA, R. H. da 2001. Introdução ao processamento de imagens ou aplicação da álgebra linear? *Revista de Matemática e Estatística*, V. 19, p. 123-144. São Paulo: Editora da UNESP. (25/06/98)
- BECK, J., STERN, M., and HAUGSJAA, E. 1996. Applications of AI in education. *Crossroads* 3, 1 (Sep. 1996), 11-15. <http://doi.acm.org/10.1145/332148.332153>
- BUNT A. and CONATI C. 2003. Probabilistic Student Modelling to Improve Exploratory Behaviour. *Journal of User Modeling and User-Adapted Interaction*, 13 (3): 269-309.
- CABRAL, T. C. B. 2005. Ensino e Aprendizagem de Matemática na Engenharia e o Uso de Tecnologia. *RENOTE – Revista Novas Tecnologias na Educação*, 3(2), nov. Porto Alegre: UFRGS, CINTED.
- CONATI, C., GERTNER, A., VANLEHN, K. 2002. Using Bayesian Networks to Manage Uncertainty in Student Modeling. *Journal of User Modeling and User-Adapted Interaction*, 12(4): 371-417.
- COWEL, R., DAWID, P.R., LAURITZEN, S. L. and SPIEGELHALTER, D. J. 1999. *Probabilistic Networks and Expert Systems*. New York: Springer-Verlag.
- CURY, H. N. 2004. Análise de Erros em Educação Matemática, In: *Veritati*, Salvador, 3(4): 95-107.
- FERNANDEZ-MANJON, B., CIGARRAN, J. M., NAVARRO, A. and FERNANDEZ-VALMAYOR, A. 1998. Using Automatic Methods for Structuring Conceptual Knowledge in Intelligent Learning Environments. In: ITS98 Conference, San Antonio, Texas, 1998. *Proceedings ...*, p. 264 – 273.
- FLORES, C.D., SEIXAS, L.J., GLUZ, J.C., PATRICIO, D., GIACOMEL, L., GONÇALVES, L. and VICARI, R.M. 2004. Amplia Learning Environment Architecture. *Revista Tecnologia da Informação*, 4(1): 27-36, nov.
- FLORES, C.D., SEIXAS, L., GLUZ, J.C. and VICARI, R.M. 2005. A Model for Pedagogical Negotiation. In: EPIA 2005 Conference, Portugal, 2005. *Lecture Notes in Computer Science*, 3008(2005): 488-499.
- GLUZ, J. C. A Biblioteca FACIL (FIPA-ACL Interface Library). 2002. 64p. *Trabalho Individual de Pesquisa*, Instituto de Informática, UFRGS, Porto Alegre.
- GLUZ, J. C. and VICCARI, R. M. 2003. Linguagens de Comunicação entre Agentes: Fundamentos Padrões e Perspectivas. In: JORNADA DE MINI-CURSOS DE INTELIGÊNCIA ARTIFICIAL, 3. Campinas, 2003. *Livro Texto*. Campinas: SBC, p. 53-102.
- GLUZ, J. C., FLORES, C. D. and VICARI, R. M. 2006. Formal Aspects of Pedagogical Negotiation in AMPLIA System. To appear in: NEDJAH, N. and MOURELLE, L. M. (eds.) *Intelligent Educational Machines*. Series: Intelligent Systems Engineering Book Series. Springer-Verlag, 2006.

- GÜRER, D. 1998. The Use of Distributed Agents in Intelligent Tutoring. In: 2nd ITS Workshop on Pedagogical Agents, San Antonio, Texas. *Proceedings*, p. 20-25.
- MARIANI, V. C. 2005. Análise de Erros em Cálculo Diferencial e Integral nos Cursos de Engenharia. In: COBENGE 2005, Campina Grande, 2005. *Anais ...*
- MAXIMA. 2007. Maxima, a computer algebra system. <http://maxima.sourceforge.net>
- MURRAY, C., VANLEHN, K. and MOSTOW, J. 2001. A Decision-Theoretic Architecture for Selecting Tutorial Discourse Actions. In: AIED-2001 Workshop on Tutorial Dialogue Systems, 2001. *Proceedings ...*
- PEARL, J. Belief Networks Revisited. 1993. *Artificial Intelligence*, Amsterdam, 59: 49-56.
- PIAGET, J., 1970. *The book, L'Épistémologie Génétique*, Paris.
- SANDHOLM, T. W. 1999. Distributed Rational Decision Making. In: WEISS, G. (ed.). *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. Cambridge: The MIT Press, p.79-120.
- SCHWARZ, B.B., NEUMAN, Y., GIL, J. and ILYA, M. 2001. Effects of argumentative activities on collective and individual arguments. In: European Conference on Computer-Supported Collaborative Learning – Euro-CSCL 2001, Maastricht, 22 - 24 March 2001. *Proceedings ...*
- SEIXAS, L.J., VICARI, R.M. and FAGUNDES, L.C. 2006. Pedagogic Strategies Based on the Student Cognitive Model Using the Constructivist Approach. To appear in: NEDJAH, N. and MOURELLE, L. M. (eds.) *Intelligent Educational Machines*. Series: Intelligent Systems Engineering Book Series. Springer/Nova, 2006.
- SELF, J. *Computational Viewpoints*. 1992. In: MOYSE, R. and ELSOM-COOK, M. (eds.) Knowledge Negotiation. Paul Chapman, London, p. 21-40.
- SELF, J. A. 1998. Hanging by Two Threads: The Evolution of Intelligent Tutoring Systems Research. In: ITS98 Conference, San Antonio, Texas, 1998. *Proceedings ...*
- VICARI, R.M., FLORES, C.D., SILVESTRE, A.M., SEIXAS, L.J., LADEIRA, M. and COELHO, H. 2003. A multi-agent intelligent environment for medical knowledge. *Artificial Intelligence in Medicine*, Elsevier, 27: 335-366.