Towards Computer-Assisted Proof Tutoring*

Marvin Schiller[1], Dominik Dietrich[1], and Christoph Benzmüller[1,2]

[1]Dept. of Computer Science, Saarland University,
66123 Saarbrücken, Germany
[2]Computer Laboratory, The University of Cambridge,
Cambridge, CB3 0FD, UK
{schiller, dodi, chris}@ags.uni-sb.de

July 31, 2007

Abstract

We present a recent application area of the proof assistant ΩMEGA, the teaching of mathematical proofs within an environment for tutorial dialog. We discuss the design of our dialog system prototype for proof tutoring in the light of the requirements imposed by its potential users. Empirical studies investigating those requirements guide the development of the system.

1 Introduction

We present work in the DIALOG [BFG+03] project, which addresses the question how the automated teaching of mathematical proof techniques can be supported with the help of a mathematical assistant system, ΩMEGA [SBA06]. The DIALOG project relies on the assumption that tutorial dialog is an effective means for teaching, and develops a prototype of such a dialog system. We investigate the needs imposed on such a system by their users with the help of empirical experiments and the gradual development of the prototype. Our work relies on the following assumptions:

1. Teaching mathematical proofs by textbooks or human tutors generally involves the use of both natural language and formula language, we therefore aim at a system capable of communication in this language;
2. The dialog system is designed to offer interactive proof exercises to their users. Learners have high expectations w.r.t. the feedback of the system to their proof attempts, which requires the system to thoroughly analyze the user input;

*Funded by the DFG SFB378 (Project DIALOG), by EPSRC under grant EP/D070511/1 and by a grant from *Studienstiftung des Deutschen Volkes e.V.*

3. Mathematics, in particular theorem proving, allows a large number of valid alternatives to the same problems, which calls for the dynamic evaluation of a particular proof attempt;
4. Users have different learning histories and requirements, and therefore require individual modeling.

This paper is organized as follows: In Section 2, we motivate and discuss our standpoint. In Section 3, we present the two main aspects of our work; empirical studies in the domain of tutorial dialog for mathematical proofs; and the prototype dialog system developed in cooperation between the DIALOG project and the ΩMEGA group.

2 Assertion Application for Proof Tutoring

The dialog system outlined above relies on a mathematical domain-reasoning component in the form of a mathematical assistant system. Research in theorem proving has resulted in a number of various systems which guarantee the correctness of proofs constructed with them. A number of such systems allow efficient proof search, but this comes at the cost that the "machine-oriented" proofs typically produced by these systems do not resemble proofs as they are tought in classrooms.

This has motivated us to consider an alternative to "classical" theorem proving, namely assertion-level theorem proving. A proof on the assertion level (a notion due to Huang [Hua94]) is a proof where each inference step represents the application of a mathematical fact, such as a definition, lemma or a theorem. Automated proof search at the assertion level is supported by the ΩMEGA system.

Besides our aim to reach a close correspondence between mathematical practice and its formal counterpart, we want to enable the dialog system to communicate with the user in a natural way. We support the development of the system with empirical experiments. These do not only highlight the requirements for the system to be effective, they also show the influence of teaching style on the learners. The following sections present these empirical studies in more detail, and discuss the design of the ΩMEGA system and the DIALOG prototype system.

3 Towards Tutorial Dialogue for Mathematical Proofs

A central role of the envisioned dialog system is to allow a learner to do interactive proof exercises with the system and receive feedback, help and hints. Therefore, the system needs to meet a number of requirements, concerning usability and its natural language interface, but also the quality of feedback. A central role of the mathematical assistant system ΩMEGA within the tutoring scenario is to analyze proof attempts from the students, in order to generate accurate feedback. This task is referred to as "Proof Step Evaluation", and a

prerequisite for effective tutoring. We have studied how human tutors perform this analysis in two series of Wizard-of-Oz experiments, where human tutors simulated the envisioned dialog system. This allowed us to collect large corpora that document the interaction between the subjects of the experiment, who acted as learners, and the simulated dialog system.

3.1 Empirical Studies

Two series of Wizard-of-Oz experiments ([WVT+04],[BHL+06a]) have been carried out in the DIALOG project. Both series served to collect corpora of tutorial dialogs, and to point out requirements to the system. In order to obtain a valid sample of mathematical teaching practice, the human experts acting as "wizards" all had teaching experience in mathematics, and the subjects in the role of the learner were university students with a basic university-level mathematics background.

During the experiment sessions, the students were asked to solve proof exercises in collaboration with the dialog system (without being informed of the wizard in a separate room). They were given preparatory material illustrating the mathematical domain, and in the first series of experiments, they had to complete a pre-test on paper. The dialogs between the wizard and the student was mediated by a software interface [BHL+06b], which included user interfaces for the wizard and the student, and recorded the tutorial dialogs in a logfile. The user interfaces allowed mixed formula and text input. In the second series of experiments, the interface was capable of different ways to input symbols, by means of selection from menu buttons, by using latex commands, and by using a German language version of latex commands.

Besides the dialogs as such, we collected additional data on the usage of the system, including video and audio recordings. Each student also had to fill out two questionnaires, one in the beginning of the experiments - asking the student about general background and previous experiences with mathematics - and one at the end, which referred to the usability of the system, but also included open-ended questions.

3.2 First Observations and Results

We have obtained a very diverse corpus of dialogs, including different patterns of using natural language, and different mathematical approaches, but which also reflect different levels of mathematical ability. In the second series of experiments, two different kinds of introductory material were handed out to two groups of approximately half of the students, which allowed us to verify if these materials had any influence on the dialogs. Indeed, it appears that the style of the introductory material had an influence on the natural language verbosity of the students (cf. [BHL+06a]). Another study [Sch05] shows a possible influence of the study material on the mathematical style of the students. Valid solutions to the proof exercises included proofs which were based on a pure "rewriting" style to proving, whereas others employed the extensionality principle and case

st[0]:	$(R \circ S)^{-1} = \{(y,x)\|(x,y) \in (R \circ S)\}$	st[0]:	One needs to show equality between two sets.
tu[0]:	This statement is correct.	tu[0]:	That's right! How do you proceed?
st[1]:	$(R \circ S)^{-1} = \{(y,x)\|\exists z(z \in M \wedge (x,z) \in R \wedge (z,y) \in S\}$	st[1]:	I use the extensionality principle.
tu[1]:	This formula is also correct.	tu[1]:	That's right.
st[2]:	$(R \circ S)^{-1} = \{(y,x)\|\exists z(z \in M \wedge (z,x) \in R^{-1} \wedge (y,z) \in S^{-1}\}$	st[2]:	Let $(s,r) \in (R \circ S)^{-1}$. According to the definition of the inverse relation it then holds that $(r,s) \in (R \circ S)$.
tu[2]:	This is correct. You are on a good way.	tu[2]:	That's right!

Figure 1: Two dialog fragments from the corpus for the proof exercise: $(R \circ S)^{-1} = S^{-1} \circ R^{-1}$, exhibiting a rewriting style (left) and the extensionality principle (right).

splits. An example of two dialog fragments (containing three interactions each) in these two different styles is represented in Figure 1. The choice of one of these two styles bears some correlation to the instruction materials given to the students.

Not only the introductory material, also the user interface exerted an influence on the dialogs. In the second series of experiments, students were shown how to use the copy & paste facilities of the interface, which lead to some striking examples of redundancy and inflated formulae during the dialogs.

The experiments illustrate the influence that the tools for learning have on their user group. Therefore, also the effect of computer systems for education on the learners needs to be carefully evaluated empirically. The experiments also show great varieties w.r.t. different users and their abilities, which calls for user modeling and the dynamic evaluation of user input.

3.3 System Design

Considering the various tasks of the envisioned dialog system mentioned above, we partition these into five modules.

Language analysis, accepting a mixture of both natural language and formula language from the user. The series of empirical experiments have allowed the study of particular idiosyncracies of mathematical texts, and a number of linguistic phenomena have been identified (see [HW06]).

Domain reasoning, based on ΩMEGA, to analyze proof steps proposed by the students, and capable of generating possible continuations for a proof, in case the student requires a hint.

Didactic knowledge, to determine what teaching strategy to follow (e.g. whether to follow a didactic or a socratic approach to teaching).

Feedback realization, which outputs a textual representation of the appropriate feedback generated with the help of the above components.

Dialog management, to orchestrate the above processes within the dialog system architecture.

These modules are arranged in a star-shaped architecture, where each module communicates directly with the dialog manager (see [BB06]). In contrast to a pipeline architecture, this allows the interleaving of the processes in different modules. This way, we enable a successively refined language analysis based on results from the domain reasoning component, or an interaction between didactic and domain reasoning components. The domain reasoning module has a pivotal role in the architecture, since it provides necessary input for the didactic module and feedback generation.

3.4 Domain Reasoning

The proof step evaluation performed by the module determines three criteria, whether a proof step is correct, whether the step size of the individual proof steps is appropriate (a.k.a. "granularity"), and whether a proof step is relevant. Evaluating these three criteria is motivated by the empirical experiments, which showed that correctness is not the only criterion that determines the feedback of the tutors. Consider the following sample from the dialogs of the second series of experiments.

st[1]	$(x, y) \in (R \circ S)^{-1}$
tu[1]	Now try to draw inferences from that!
st[2]	$(x, y) \in S^{-1} \circ R^{-1}$
tu[2]	One cannot directly deduce that. You need some intermediate steps!

As the basis for proof step evaluation, each proof step proposed by the user is reconstructed in ΩMEGA. Since even the most ordinary human proof steps can generally include a number of tacit intermediate steps, which become apparent when modeling these proof steps in a rigorous formal system, the reconstruction requires proof search. The details of proof reconstruction in ΩMEGA are described in [DB07]. It delivers a proof object at the assertion level, i.e., a proof where each step is justified by a mathematical fact such as a definition, a lemma or a theorem, which is a formal (and verified) model of the originally uttered proof step. In case a proof cannot be found for a given utterance within reasonable resource bounds, the proof step is considered incorrect.

3.5 Adapting Proof Step Analysis to Empirical Norms – The Case of Granularity

The ΩMEGA proof reconstructions at the assertion level also allow us to measure the step size of proof steps. It can be argued that they provide a better

approximation to human step size than many other calculi which are popular in theorem proving (even those calculi that were invented with mathematical practice in mind, such as Gentzen's Natural Deduction calculus [Gen34]). Consider, for example, two proof fragments (one using ΩMEGA assertion application, and one in Natural Deduction) which both reconstructs the same example of an uttered proof step from the experiments (namely, that $(y,z) \in r \wedge (x,z) \in s^{-1}$ follows from $(z,y) \in r^{-1} \wedge (x,z) \in s^{-1}$, where r^{-1} denotes the inverse of the relation r).

$$\frac{(z,y) \in r^{-1} \wedge (x,z) \in s^{-1}}{(y,z) \in r \wedge (x,z) \in s^{-1}}\ Def^{-1}$$

Assertion Application

$$\dfrac{\dfrac{\dfrac{A := (z,y) \in r^{-1} \wedge (x,z) \in s^{-1}}{(y,z) \in r^{-1}}\ \wedge E}{(y,z) \in r}\ Def.^{-1} \qquad \dfrac{A}{(x,z) \in s^{-1}}\ \wedge E}{(y,z) \in r \wedge (x,z) \in s^{-1}}\ \wedge I$$

Natural Deduction

However, the experiments suggest that step size is not an absolute quantity, but depends on the context of a tutorial dialog – for an advanced student, an appropriate step size can be much larger than for a beginner. In order to account for this, we include a student model into the system architecture. It records whenever a particular mathematical fact (associated to concepts, like particular definitions or theorems) has been applied successfully, and allows the analysis to distinguish between concepts the student is acquainted to, and concepts the student is not expected to know. This way, granularity analysis can be parameterized depending on the knowledge of the student, which dynamically changes during the session.

The experiments also provide evidence that different tutors had different opinions of what constitutes an appropriate step size. However, the dialogs presented to them were not directly controlled by the setup of the experiments, we can only give a phenomenological account of these differences. Nevertheless, this highlights the requirement for the granularity analysis to allow different viewpoints on granularity. These naturally arise from a number of different criteria of a proof step that might play a role for judging step size, such as

- Does a proof step involve one or even more concepts that are considered unknown to the student?
- Does a proof step introduce hypotheses or a case split?
- Are concepts required in a proof step mentioned verbaly?
- Do the applied facts have the status of definitions, theorems or lemmata?
- The total number of different concepts required for making a particular step.

This is certainly not exhaustive, but it illustrates that a teacher has much freedom in chosing to what degree each of these criteria influences his judgment w.r.t. step size.

3.6 Discussion and Outlook

This paper has presented arguments why in the field of computer-aided mathematics tutoring, a strong coupling of system development and empirical studies is beneficial. We have illustrated this process w.r.t. the DIALOG project, where the development is gradually geared to the needs of potential users. This has revealed a number of parameters that have to be taken into account, including the particularities of mathematical language, use of the interface, a variety of approaches to proving (and in particular, a gap between human modes of proving compared to computer systems), and individual differences among users, but also among tutors, some of which were outlined in Section 3.2. This has stimulated the current conception of the DIALOG system, and further motivated the choice of ΩMEGA as a domain reasoner.

Future plans include the deployment of a subset of the developed modules and techniques from the prototype in the e-learning platform Activemath [MMU+07], as well as further case studies with the emerging prototype system, which is currently still under development.

References

[BB06] Mark Buckley and Christoph Benzmüller. An Agent-based Architecture for Dialogue Systems. In Irina Virbitskaite and Andrei Voronkov, editors, *Proceedings of Perspectives of System Informatics*, volume 4378 of *Lecture Notes in Computer Science*, pages 135–147, Novosibirsk, Russia, 2006. Springer.

[BFG+03] Christoph Benzmüller, Armin Fiedler, Malte Gabsdil, Helmut Horacek, Ivana Kruijff-Korbayová, Manfred Pinkal, Jörg Siekmann, Dimitra Tsovaltzi, Bao Quoc Vo, and Magdalena Wolska. Tutorial dialogs on mathematical proofs. In *Proceedings of the IJCAI Workshop on Knowledge Representation and Automated Reasoning for E-Learning Systems*, pages 12–22, Acapulco, 2003.

[BHL+06a] Christoph Benzmüller, Helmut Horacek, Henri Lesourd, Ivana Kruijff-Korbajova, Marvin Schiller, and Magdalena Wolska. A corpus of tutorial dialogs on theorem proving; the influence of the presentation of the study-material. In *Proceedings of International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy, 2006. ELDA.

[BHL+06b] Christoph Benzmüller, Helmut Horacek, Henri Lesourd, Ivana Kruijff-Korbajova, Marvin Schiller, and Magdalena Wolska. DiaWOz-II - A Tool for Wizard-of-Oz Experiments in Mathematics. In *Proceedings of the 29th Annual German Conference on Artificial Intelligence*, Lecture Notes in Computer Science. Springer-Verlag, 2006.

[DB07] Dominik Dietrich and Mark Buckley. Verification of proof steps for tutoring mathematical proofs. In *Proceedings of AIED 2007*, 2007.

[Gen34] Gerhard Gentzen. Untersuchungen über das logische Schliessen. *Mathematische Zeitschrift*, 39:176–210, 405–431, 1934.

[Hua94] X. Huang. Reconstructing proofs at the assertion level. In A. Bundy, editor, *Automated Deduction-CADE-12*, pages 738–752. Springer, Berlin, Heidelberg, 1994.

[HW06] Helmut Horacek and Magdalena Wolska. Interpreting semi-formal utterances in dialogs about mathematical proofs. *Data & Knowledge Engineering*, 58(1):90–106, 2006.

[MMU+07] Erica Melis, Marianne Moormann, Carsten Ullrich, Georgi Goguadze, and Paul Libbrecht. How activemath supports moderate constructivist mathematics teaching. In *8th International Conference on Technology in Mathematics Teaching*, Hradec Kralove, 2007.

[SBA06] Jörg H. Siekmann, Christoph Benzmüller, and Serge Autexier. Computer supported mathematics with omega. *J. Applied Logic*, 4(4):533–559, 2006.

[Sch05] Marvin Schiller. Mechanizing Proof Step Evaluation for Mathematics Tutoring - The Case of Granularity. Master's thesis, Universität des Saarlandes, 2005.

[WVT+04] M. Wolska, B. Quoc Vo, D. Tsovaltzi, I. Kruijff-Korbayová, E. Karagjosova, H. Horacek, M. Gabsdil, A. Fiedler, and C. Benzmüller. An annotated corpus of tutorial dialogs on mathematical theorem proving. In *Proceedings of International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, Portugal, 2004. ELDA.