# Judging Granularity for Automated Mathematics Teaching[⋆]

Marvin Schiller, Christoph Benzmüller, and Ann Van de Veire

Dept. of Computer Science
Saarland University, 66041 Saarbrücken, Germany
{schiller, chris, veire}@ags.uni-sb.de

**Abstract.** In proof tutoring, human maths tutors are observed to reject correct proof steps if they are not at the expected level of granularity, i.e. if they are too detailed or too coarse-grained. We investigate how the judgments on granularity as observed from human tutors can be automated with the help of automated reasoning techniques. We evaluate our approach with data collected in an empirical study.

**Key words:** automated reasoning, proof tutoring, theorem proving

## 1 Introduction

The use of automated theorem provers within tutoring environments for mathematical proofs poses a challenge. Didactic considerations require theorem provers to support actual mathematical practice, in addition to providing powerful automation in a selected mathematical domain. Since the development of classical theorem provers and investigations on logical calculi are mainly driven by correctness, completeness and efficiency issues, these theorem provers can generally not be used to determine the general appropriateness of a proof step proposed by a learner in a tutorial context. For instance, human maths teachers are known to reject proof steps that are logically correct if they lack other desirable properties, in particular, if the steps are of inappropriate step size or irrelevant. This motivates the development of techniques suitable for the automated analysis of proof steps w.r.t. their *correctness*, but also their *granularity*, i.e. the argumentative size of a proof step, and their *relevance* in reaching the goal of a proof.

In this paper we report on ongoing work to automate the analysis of the granularity of proof steps in the frame of the Dialog project [1][1]. The Dialog project investigates natural tutorial dialog between a student and a mathematical assistant system. At the present time, the (simplified) approach for the Dialog system is:

1. The student inputs a proof via the user interface using a mixture of natural language and formulas. The analysis of this input and its conversion into a

[⋆] This work has been funded by the DFG Collaborative Research Center on Resource-Adaptive Cognitive Processes, SFB 378 (http://www.coli.uni-saarland.de/projects/sfb378/).

[1] http://www.ags.uni-sb.de/~dialog/

formalized representation suitable for a mathematical assistant system is a challenge for computational linguistics.

2. Automated reasoning techniques are employed to support the tutoring of mathematical proofs. The Dialog project relies on the mathematical assistant system $\Omega$mega developed in the $\Omega$mega group[2]. In particular, the $\Omega$mega environment is employed to analyze the student's formalized proof step in order to judge its *soundness*, its *granularity* and its *relevance* for solving the proof problem.
3. The analysis results of step (2) are passed on to a tutoring module which determines appropriate feedback to be presented to the student via the Dialog system user interface. The tutoring module may employ further relevant information, such as a learner and a teacher model.

In summer 2005, the Dialog project collected a corpus of tutorial dialogs, in which thirty-seven students interacted with a mock-up dialog system, simulated with the help of four experienced human tutors in turn. The students were given exercises in the domain of binary relations, dealing with the properties of the inverse relation and relation composition, which the students had to solve in a collaborative dialog with the system (for details, see [2]). In order to investigate the role of granularity, we asked the tutors to annotate each proof step proposed to them by the students with one out of three granularity categories; namely *appropriate*, *too detailed* and *too coarse-grained*. On average, 1.92 student utterances per session were classified by the experts as either *too detailed* or *too coarse-grained*, out of an average of 25 utterances per session.

Related work has also identified granularity as a worthwhile study subject for the design of e-learning systems, under the aspect of cognitive principles [3] and also in the specific context of a tutoring system for LISP programming [4]. Another work in the domain of mathematical proof illustrates the schism that exists between the step size in classical theorem proving and in human practice. McMath et al. [5] integrate the theorem prover *Otter* into an environment for mathematics in order to automatically verify subproofs of a proof under construction. In order to prevent the resolution-calculus based *Otter* prover from allowing "too large leaps of logic", they limit the use of *Otter* to a few seconds, a strategy which they observe to be unsatisfactory. In particular, in case the time limit prevents *Otter* from finding a proof, it remains unclear whether this is because the subproof represents a too big step, or whether the subproof is incorrect, or whether the employed proof search strategy is inappropriate.

## 2 Techniques and Calculi for Granularity Analysis

We present a procedure that is aimed at judging the granularity of a user's proof step in a proof attempt, and evaluate it with data from the experiment reported in [2]. The work is described in detail in [6]. Then we describe work in progress which is aimed at refining the approach.

[2] `http://www.ags.uni-sb.de/~omega`

### 2.1 Granularity Relative to a Proof Calculus

The problem we are investigating is the following: given the formalized version of a proof step proposed by a student, how can we estimate its argumentational complexity? In the setting of the Dialog project, we allow *underspecified* input, such that a student may claim that a formula $A$ follows from some other formulas $B_1$, $B_2$, etc... without indicating which rules of inference were used and which assertions were implicitly assumed. In particular, the user is not restricted to a fixed set of inference rules, but may simply state that $A$ follows from $B_1$, $B_2$, etc... . Whether this proposed inference is correct can be checked with automated reasoning techniques, however, on the level of a formal calculus, this usually requires a number of intermediate steps. An example proof step from the experiment described above is[3]:

(1) student: [...] $(z, y) \in R^{-1}$ and $(x, z) \in S^{-1}$
tutor: [Correct.]
student: It follows: $(x, y) \in S^{-1} \circ R^{-1}$

If we look at this proof step formally, it consists of both the application of the standard definition of $\circ$ and the commutativity of $\wedge$. Generally, a single proof step from the user can require $n$ steps in a formal calculus.

We now consider a simple procedure which uses automated reasoning to build a minimal reconstruction of the proof step at the level of a formal proof system $\vDash$, and then merely counts the number of inference steps in the reconstruction. We then use this number as an indicator for the argumentative complexity of the original proof step. The approach relies heavily on the assumption that a calculus exists which is both suitable for automation and at the same time reflects the step size of human mathematical reasoning. As a first step, we tested this hypothesis with two natural deduction calculi, the traditional natural deduction (ND) calculus by Gentzen [7] and the empirically motivated PSYCOP (shorthand for "Psychology of Proof") calculus [8]. Before considering any further options, such as assertion level proofs [9], we decided to evaluate their suitability first. Besides the choice of a suitable calculus, a second question concerns how to extract a measure for granularity from the constructed proof object. The simple approach of summing up the number of calculus rule applications assumes an equal contribution of all inference rules to the complexity of a proof fragment. A more elaborate approach consists in building weighted sums over the inference rule applications. In order to account for these options, we have implemented a framework for granularity analysis which is parameterized over the choice of a concrete calculus and a particular weighting (for details see [6]).

We performed a preliminary evaluation of our approach with a sample of twenty proof steps from the experiment in the Dialog project. The proof steps were formalized, and then analyzed with the described framework, using both Gentzen's ND calculus and PSYCOP as the proof calculi, for comparison. The

[3] We denote the inverse relation of $R$ as $R^{-1}$, and we denote the binary operator for relation composition by $\circ$.

**Table 1.** Average number of calculus level proof steps that constitute a student's proof step for twenty steps from the study, grouped by their granularity level as identified by the tutors. The "too detailed" group consists of only two student proof steps, therefore standard deviations are omitted. Calculus level proofs of zero length usually occur when the formalization of the analyzed statement is identical to a previous statement by the user.

| Tutor's rating | Avg. proof step length at calculus level (with std. deviation) | |
|---|---|---|
| | PSYCOP calculus | Gentzen's ND calculus |
| "too detailed" | 1,00 | 0 |
| "appropriate" | 5,27 (4,88) | 5,00 (5,14) |
| "too coarse-grained" | 11,67 (6,80) | 10,33 (7,72) |

complexity of a proof step was assumed to be represented simply by the number of calculus level steps obtained by reconstructing the original proof step in Gentzen's ND and the PSYCOP calculus. The PSYCOP theory [8] explicitly provides a decision procedure for proof search, which we followed in the evaluation. In the case of Gentzen's ND, we considered the shortest derivation in the number of calculus level proof steps to be representative of the original proof step.

### 2.2 First Results

The results (cf. Table 1) show indeed a tendency of those proof steps that were *too detailed* in the eyes of the tutor to require shorter reconstructions on the calculus level than the average *appropriate* proof step, which again on average required shorter reconstructions than those steps that were *too coarse-grained.* This would indeed support the hypothesis that granularity is reflected in the number of proof steps on the level of the two natural deduction calculi. However, this method alone appears to be insufficient to distinguish between the three granularity classes *too detailed*, *appropriate* and *too coarse-grained* from the experiment, which is the aim of the granularity analysis. As indicated by the corresponding standard deviations, the sizes of the proof step reconstructions vary greatly within the *appropriate* and the *too coarse-grained* group. The comparison between using Gentzen's ND and PSYCOP on this particular sample of proof steps shows no significant difference.

## 3 Discussion

Our goal of mechanizing the analysis of granularity judgments has led us to evaluate the suitability of human-oriented calculi and proof mechanisms for this purpose. A first evaluation on a specific sample of proof steps shows no advantage of one of the two considered natural deduction calculi over the other. This preliminary evaluation motivates the investigation of other candidate calculi like the Core [10] calculus, which has recently been implemented as a part of the $\Omega$mega-Core framework. In particular, we observed that the students in our experiments often used proof steps that can be characterized as rewriting steps or deep inference steps, which are more naturally represented in CORE than in the two considered natural deduction calculi. Thus, we expect that the granularity analysis measures we can obtain with CORE as base calculus will be even more convincing.

Furthermore, in the above evaluation we assume that granularity judgments by the mathematics experts depend only on the purely logical properties of a given proof step, related to its calculus level proof. However, personal preferences of the involved experts, the maths proficiency of each particular student and the context of each proof step were ignored completely, even though they can provide further explanation. Our ongoing work aims at guiding the theorem-prover-based granularity analysis with information from a student model, a tutoring model and an annotated repository of the mathematical domain, and enhancing it with an expert system that decides on the appropriateness of the proof step in question. This alters the original conception of the three-step approach in the DIALOG system presented in Section 1, such that not only the tutoring module interacts with a student and a tutoring model, but also the proof step analysis. We conclude that beyond automated reasoning on the mathematical domain level, our investigation motivates the development of techniques that support the reasoning *about* the appropriateness of a particular proposed proof step.

## References

1. Benzmüller, C., Horacek, H., Kruijff-Korbayová, I., Pinkal, M., Siekmann, J., Wolska, M.: Natural language dialog with a tutor system for mathematical proofs. Journal of Computer Science and Technology (2006). To appear.
2. Benzmüller, C., Horacek, H., Lesourd, H., Kruijff-Korbayová, I., Schiller, M., Wolska, M.: 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, ELDA (2006)
3. Anderson, J.R., Boyle, C.F., Farrell, R., Reiser, B.J.: Cognitive principles in the design of computer tutors. In Morris, P., ed.: Modelling Cognition. Wiley (1987)
4. McCalla, G., Greer, J., Barrie, B., Pospisel, P.: Granularity hierarchies. In Lehmann, F., ed.: Semantic Networks in Artificial Intelligence. Pergamon Press, Oxford (1992) 363–375
5. McMath, D., Rozenfeld, M., Sommer, R.: A computer environment for writing ordinary mathematical proofs. In Nieuwenhuis, R., Voronkov, A., eds.: LPAR. Volume 2250 of Lecture Notes in Computer Science, Springer (2001) 507–516
6. Schiller, M.: Mechanizing Proof Step Evaluation for Mathematics Tutoring - the Case of Granularity. Master's thesis, Universität des Saarlandes, Germany (2005)
7. Gentzen, G.: Untersuchungen über das logische Schliessen. Mathematische Zeitschrift **39** (1934) 176–210, 405–431
8. Rips, L.J.: The psychology of proof : deductive reasoning in human thinking. MIT Press, Cambridge, MA (1994)
9. Huang, X.: Human Oriented Proof Presentation: A Reconstructive Approach. Phd thesis, Universität des Saarlandes, Germany (1994)
10. Autexier, S.: The core calculus. In Nieuwenhuis, R., ed.: CADE. Volume 3632 of Lecture Notes in Computer Science, Springer (2005) 84–98