Granularity-Adaptive Proof Presentation[1]

Marvin Schiller [a,b] and Christoph Benzmüller [c]
[a] *German Research Center for Artificial Intelligence (DFKI), Bremen, Germany*
[b] *Saarland University, Saarbrücken, Germany*
[c] *International University in Germany, Bruchsal, Germany*

1. Introduction

Granularity matters in mathematics. For example, in introductory textbooks intermediate proof steps are often skipped, when this seems appropriate. Such a situation is given in the elementary proof in basic set theory reproduced in Figure 1. Whereas most of the proof steps consist of the application of exactly one mathematical fact (a definition or a lemma), the step from assertion [9] to assertion [10] applies several inference steps at once, namely the application of the definition of $\cap$ twice, and the distributivity of *and* over *or*. Similar observations were made in the empirical studies within the DIALOG project (cf. [8]).

Systems like ΩMEGA [1] and HiProofs [4] are capable of structuring proofs hierarchically, the problem remains though how to identify a suitable level of granularity. Autexier and Fiedler have proposed one particular level of granularity [2], which they call *what-you-need-is-what-you-stated granularity.* Their rigid solution, however, fails to fully model the proof in Figure 1.

We present a flexible approach to proof presentation that dynamically adapts to specific levels of granularity in context. Different models for granularity can be learned in our framework from samples using machine learning techniques. More information on the work sketched here is available in a technical report [8].

[1] This work was supported by a grant from *Studienstiftung des Deutschen Volkes e.V.*

[1] Let x be an element of $A \cap (B \cup C)$, [2] then $x \in A$ and $x \in B \cup C$. [3] This means that $x \in A$, and either $x \in B$ or $x \in C$. [4] Hence we either have (i) $x \in A$ and $x \in B$, or we have (ii) $x \in A$ and $x \in C$. [5] Therefore, either $x \in A \cap B$ or $x \in A \cap C$, so [6] $x \in (A \cap B) \cup (A \cap C)$. [7] This shows that $A \cap (B \cup C)$ is a subset of $(A \cap B) \cup (A \cap C)$. [8] Conversely, let y be an element of $(A \cap B) \cup (A \cap C)$. [9] Then, either (iii) $y \in A \cap B$, or (iv) $y \in A \cap C$. [10] It follows that $y \in A$, and either $y \in B$ or $y \in C$. [11] Therefore, $y \in A$ and $y \in B \cup C$ so that $y \in A \cap (B \cup C)$. [12] Hence $(A \cap B) \cup (A \cap C)$ is a subset of $A \cap (B \cup C)$. [13] In view of Definition 1.1.1, we conclude that the sets $A \cap (B \cup C)$ and $(A \cap B) \cup (A \cap C)$ are equal.

Figure 1. Proof of the statement $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$, reproduced from [3].

8. We assume $y \in (A \cap B) \cup (A \cap C)$ and show $y \in A \cap (B \cup C)$
9. Therefore, $y \in A \cap B \vee y \in A \cap C$
10.(a) Therefore, $y \in A \wedge B \vee y \in A \cap C$
10.(b) Therefore, $y \in A \wedge B \vee y \in A \wedge C$
10.(c) Therefore, $y \in A \wedge (y \in B \vee y \in C)$
11. Therefore, $y \in A \wedge y \in B \cup C$

(a)

8. We assume $y \in (A \cap B) \cup (A \cap C)$ and show $y \in A \cap (B \cup C)$
9. Therefore, $y \in A \cap B \vee y \in A \cap C$
10. Therefore, $y \in A \wedge (y \in B \vee y \in C)$
11. Therefore, $y \in A \wedge y \in B \cup C$

(b)

_ $\Rightarrow$ step-appropriate

(c)

1) conceptsunique $\in \{0, 1\}$ $\wedge$ equalitydefn=0 $\wedge$ verb=true $\Rightarrow$ step-too-small
2) hypintro=0 $\wedge$ equalitydefn=0 $\wedge$ $\cup$-Defn=0 $\wedge$ verb=true $\Rightarrow$ step-too-small
3) conceptsunique $\in \{2, 3, 4\}$ $\wedge$ $\cup$-Defn $\in \{1, 2, 3\}$ $\Rightarrow$ step-too-big

$\vdots$

_ $\Rightarrow$ step-appropriate

(d)

Figure 2. Proof fragment (a) is obtained with rule set (c) and fragment (b) with rule set (d).

2. An Adaptive Model for Granularity

In our approach proofs are initially represented at ΩMEGA's assertion level[2], and we treat the granularity problem as a classification task: given a proof step, representing one or several assertion applications, we judge it as either *appropriate*, *too big* or *too small*. As our feature space we employ several mathematical and logical aspects of proof steps as well as cognitive aspects. For example, we keep track of the background knowledge of the user in a student model.

We express our models for classifying granularity as rule sets, which associate specific combinations of feature values to a corresponding granularity verdict ("appropriate", "too big" or "too small"). These rule sets may be hand-authored by an expert or they may be learned from empirical data. Our algorithm for granularity-adapted proof presentation takes two arguments, a granularity rule set and an ΩMEGA assertion level proof tree, and it then incrementally categorizes the to-be-presented proof steps in the proof tree using the rules. We thereby obtain a proof tree with labeled proof nodes: the nodes are either categorized as appropriate or too fine-grained. Entire proof presentations are then generated by walking through the tree, skipping the too fine-grained steps.[3]

Case Study We exemplarily model the step size of the textbook proof in Figure 1, starting with an assertion-level proof presentation consisting of 15 steps in ΩMEGA, and skipping intermediate proof steps according to our feature-based granularity model. Figure 2 (a) shows a proof fragment which corresponds to steps 8–11 in Figure 1 and which was generated from our initial ΩMEGA assertion level proof with the trivial rule set in Figure 2 (c). Using the alternative rule set presented in extracts in Figure 2 (d) (with nine rules altogether) we can generate the proof as presented in Figure 2 (b). This more appropriate rule set was learned[4]. NL output is produced here via simple patterns and more exciting

[2]Assertion level proofs justify their steps by application of axioms, definitions, or theorems.

[3]Even though too fine-grained intermediate are withheld, we make sure that the presentation of the output step sufficiently reflects all intermittent assertion applications.

[4]The sample proof was used to fit the rule set to it. All steps in the sample proof were provided as *appropriate*, all intermediate assertion level steps were labeled as *too-small*, and always the next bigger step to each step in the original proof was provided as an example for a *too big* step.

NL output is easily possible. The resulting proof presentation fits the step size of the original proof in Figure 1.

Learning from Empirical Data We employ off-the-shelf machine learning tools to learn classifiers for granularity (like our rule sets) from annotated examples (*supervised* learning). In our case, an expert annotates proof steps with the labels *appropriate*, *too small* or *too big*. We initially represent these proof steps in ΩMEGA which has the advantage that relevant proof step features are computed in the background, and combined automatically with the expert's judgments as training instances for the learning algorithm. Currently, our algorithm calls the C5.0 data mining tools [7] – which support the learning of decision trees and of rule sets – to obtain classifiers for granularity. As part of an ongoing evaluation, we have conducted a study where a mathematician (with tutoring experience) judged the granularity of 135 proof steps, presented to him via an ΩMEGA-assisted environment with intermediate assertion-level steps skipped at random.

3. Conclusion

Granularity has been a challenge in AI for decades [5]. Here we have focused on adaptive proof granularity, which we treat as a classification problem, taking into account changeable information such as the user's familiarity with mathematical concepts. Using assertion level proofs as the basis for our approach has the advantage that the relevant information for the classification task is easily read off the proofs. Moreover, it eases the generation of NL proof output.

Future work consists in empirical evaluations of the learning approach. Interesting questions are: (i) what are the most useful features for judging granularity, and are they different among distinct experts and mathematical domains, (ii) what is the inter-rater reliability among different experts and the corresponding classifiers generated by learning in our framework? The resulting corpora of annotated proof steps and generated classifiers can then be used to evaluate the appropriateness of the proof presentations generated by our system.

References

[1] S. Autexier, C. Benzmüller, D. Dietrich, A. Meier, and C.-P. Wirth. A generic modular data structure for proof attempts alternating on ideas and granularity. In Kohlhase [6].

[2] S. Autexier and A. Fiedler. Textbook proofs meet formal logic - the problem of underspecification and granularity. In Kohlhase [6].

[3] R. G. Bartle and D. Sherbert. *Introduction to Real Analysis*. Wiley, 2 edition, 1982.

[4] E. Denney, J. Power, and K. Tourlas. Hiproofs: A hierarchical notion of proof tree. In *Proceedings of the 21st Annual Conference on Mathematical Foundations of Programming Semantics (MFPS XXI)*, volume 155 of *LNCS*, pages 341 – 359. Elsevier, 2006.

[5] J. R. Hobbs. Granularity. In *Proc. of the 9th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 432–435, 1985.

[6] M. Kohlhase, editor. *Proc.of MKM'05*, volume 3863 of *LNCS*. Springer, 2006.

[7] RuleQuest Research. Data mining tools see5 and c5.0. `http://www.rulequest.com/see5-info.html`, retrieved April 15, 2009.

[8] M. Schiller and C. Benzmüller. Granularity-adaptive proof presentation. Technical report, 2009. SEKI Working-Paper SWP-2009-01.