Argumentation within Deductive Reasoning

Armin Fiedler and **Helmut Horacek**[1]

Abstract. Deductive reasoning is an area related to argumentation where machine-based techniques, notably theorem proving, can contribute substantially to the formation of arguments. However, making use of the functionality of theorem provers for this issue is associated with a number of difficulties and, as we will demonstrate, requires considerable effort for obtaining reasonable results. Aiming at the exploitation of machine-oriented reasoning for human-adequate argumentation in a broader sense, we present our model for producing proof presentations from machine-oriented inference structures. Capabilities of the model include adaptation to human-adequate degrees of granularitiy and explicitness in the underlying argumentation and interactive exploration of proofs. Enhancing capabilities in all these respects, even just those we have addressed so far, does not only improve the interactive use of theorem provers, but they are essential ingredients to support the functionality of dialog-oriented tutorial systems in formal domains.

1 Introduction

Deductive reasoning is an area related to argumentation where machine-based techniques, notably theorem proving, can contribute substantially to the formation of arguments. However, making use of the functionality of theorem provers for this issue is associated with a number of difficulties and, as we will demonstrate, requires considerable effort for obtaining reasonable results.

Aiming at the exploitation of machine-oriented reasoning for human-adequate argumentation in a broader sense, we present our model for producing proof presentations from machine-oriented inference structures. Capabilities of the model include adaptation to human-adequate degrees of granularitiy and explicitness in the underlying argumentation and interactive exploration of proofs. However, this model has inherent limitations in its argumentative behavior, since arguments giving motivations or justifications on a more strategic or dynamic perspective cannot be obtained from machine-found proofs. Enhancing capabilities in all these respects does not only improve the interactive use of theorem provers, but they are essential ingredients to support the functionality of dialog-oriented tutorial systems in formal domains.

This paper is organized as follows. We first provide some background information about presentation of machine-found proofs in natural language, including empirical motivations that substantiate divergent demands for human-adequate presentations. We describe techniques for building representations meeting these psychological requirements in a formal model, comprising some kinds of proof transformation and adaptations. We illustrate the functionality of our model by discussing a moderately complex example. Finally, we sketch some limitations of our model.

2 Background

2.1 Proof Presentation in Natural Language

The problem of obtaining a natural language proof from a machine-found proof can be divided into two subproblems: First, the proof is transformed from its original machine-oriented formalism into a human-oriented calculus, which is much better suited for presentation. Second, the transformed proof is verbalized in natural language.

Since the lines of reasoning in machine-oriented calculi are often unnatural and obscure, algorithms (see, e.g., [1, 18]) have been developed to transform machine-found proofs into more natural formalisms, such as the *natural deduction (ND) calculus* [8]. ND inference steps consist of a small set of simple reasoning patterns, such as forall-elimination ($\forall x P(x)$ leads to $P(a)$) and implication elimination, that is, modus ponens. However, the obtained ND proofs often are very large and too involved in comparison to the original proof. Moreover, an inference step merely consists of the syntactic manipulation of a quantifier or a connective. [15] gives an algorithm to abstract an ND proof to an *assertion level* proof, where a proof step may be justified either by an ND inference rule or by the application of an assertion (i.e., a definition, axiom, lemma or theorem).

One of the earliest proof presentation systems was introduced by Chester [2]. Several theorem provers have presentations components that output proofs in pseudo-natural language using canned text (e.g., [3, 4]). Employing several isolated strategies, the presentation component of THINKER [5] was the first system to acknowledge the need for higher levels of abstraction when explaining proofs. PROVERB [16] expresses machine-found proofs abstracted to the assertion level and applies linguistically motivated techniques for text planning, generating referring expressions, and aggregation of propositions with common elements. Drawing on PROVERB, we have developed the interactive proof explanation system *P.rex* [7], which additionally features user adaptivity and dialog facilities. [10] presents another recently developed NLG system that is used as a back end for a theorem prover.

In order to produce reasonable proof presentations, many systems describe some complex inference steps very densely, and they leave certain classes of proof steps implicit in their

[1] University of the Saarland, P.O.Box 151150, D-66041, Saarbrücken, F.R.G.

output, for example, by abstracting from intermediate inference steps that are recoverable from inductive definitions, or by omitting instantiations of axioms. However, leaving out information on the basis of purely *syntactic* criteria, as this has been done so far, easily leads to incoherent and hardly understandable text portions. In order to get control over the inferability and comprehensibility in presenting inference steps, an explicit model is required that incorporates semantic and pragmatic aspects of communication, which is what we try to achieve by our approach.

2.2 Empirical Motivation

Issues in presenting deductive proofs, as a special case of presenting argumentative discourse, have attracted a lot of attention in the fields of psychology, linguistics, and computer science. Central insights relevant to deductive argumentation are the following:

- Logical consequences of certain kinds of information are preferably conveyed implicitly, through relying on capabilities of the audience to exploit the discourse context and default expectations.
- Human performance in comprehending deductive syllogisms varies significantly from one syllogism to another.

The study in [23] demonstrates that humans easily uncover missing pieces of information left implicit in discourse, most notably in sequences of events, provided this information conforms to their expectations in the given context. Similarly to the expectations examined in that study, which occur frequently in everyday conversations, a number of elementary and very common inferences are typically left implicit in mathematical texts, too, including straightforward instantiations, generalizations, and associations justified by domain knowledge.

Another presentation aspect is addressed by studies on human comprehension of deductive syllogisms (see the summary in [17]). These studies have unveiled considerable performance differences among individual syllogisms (in one experiment, subjects made 91% correct conclusions for modus ponens, 64% for modus tollens, 48% for affirmative disjunction, and 30% for negative disjunction). The consequences of this result are demonstrated by the elaborate essay in [24], which presents a number of hypotheses about the impacts that human resource limits in attentional capacity and in inferential capacity have on dialog strategies. These hypotheses are acquired from extensive empirical analysis of naturally occurring dialogs and, to a certain extent, statistically confirmed. One that is of central importance for our investigations says that an increasing number of logically redundant assertions to make an inference explicit are made, in dependency of how hard and important an inference is (modus tollens being an example for a hard inference which requires a more detailed illustration).

However, these crucial issues in presenting deductive reasoning are insufficiently captured by current techniques, which typically suffer from two kinds of deficits:

- A large number of easily inferable inference steps is expressed explicitly.
- Involved inferences, though hard to understand, are presented in single shots.

The first deficit suggests the omission of contextually inferable elements in the proof graph, and the second demands the expansion of compound inference steps into simpler parts. We illustrate the appearance of these deficits and measures to remedy them in the subsequent sections.

3 An Example

Throughout this paper, we will use the proof of a well-known problem, Schubert's Steamroller [22], to demonstrate the functionality of our presentation model:

Axioms:

(1) Wolves, foxes, birds, caterpillars, and snails are animals, and there are some of each of them. Also there are some grains, and grains are plants.

(2) Every animal either likes to eat all plants or all animals much smaller than itself that like to eat some plants.

(3) Caterpillars and snails are much smaller than birds, which are much smaller than foxes, which in turn are much smaller than wolves. Wolves do not like to eat foxes or grains, while birds like to eat caterpillars, but not snails. Caterpillars and snails like to eat some plants.

Theorem:

(4) Therefore there is an animal that likes to eat a grain-eating animal.

Proving that theorem (4) is based on applying given pieces of simplified real world knowledge (1) to (3).

In a nutshell, the proof runs along the following lines: Through applying axiom (2) three times, it is first derived that birds eat plants, then that foxes do not eat grains and, finally, that foxes eat the smaller grain-eating birds, the last being the witness needed to prove theorem (4).

Within the theorem proving community, the Steamroller problem is famous, because solving it requires several variables to be instantiated purposefully without having a guidance how to do this through the formulation of the theorem to be proved — it has only existentially quantified variables in it, but no constants. Until some years ago, automated theorem provers were unable to apply this technique with sufficient degrees of efficiency, so that they were originally unable to solve this problem. For our purposes, this problem is attractive for completely different reasons: its definition is easily comprehensible without mathematical knowledge, and a full-detailed solution path is sufficiently complex so that exploring it interactively seems to be well motivated.

4 Our Model of Argument Building

In order to meet the deficits identified when discussing empirical motivations, we propose the application of an optimization process that enhances an automatically generated proof at the assertion level. Through this process, pragmatically motivated expansions, omissions, and short-cuts are introduced, and the audience is assumed to be able to mentally reconstruct the details omitted with reasonable effort. In a nutshell, the modified proof graph is built through two subprocesses:

- *Building expansions*
  Compound assertion level steps are expanded into elementary applications of deductive syllogisms, while marking the original larger steps as summaries.
- *Introducing omissions and short-cuts*
  Shorter lines of reasoning are introduced by skipping individual reasoning steps, through omitting justifications (marked as inferable) and intermediate reasoning steps (marking the 'indirect' justifications as short-cuts).

4.1 Levels of Abstraction

The purpose underlying the expansion of assertion level steps is to decompose presentations of complex theorem applications or involved applications of standard theorems into easier comprehensible pieces. This operation is motivated by performance difficulties humans typically have in comparable discourse situations. At first, assertion level steps are completely expanded to the natural deduction (ND) level according to the method described in [15]. Thereafter, a partial recomposition of ND steps into inference steps encapsulating the harder comprehensible deductive syllogisms, modus tollens and disjunction elimination steps, is performed, in case the sequence of ND rules in the entire assertion level step contains more than one of these. To do this, the sequence of ND rules is broken after each but the last occurrence of a modus tollens or disjunction elimination, and the resulting subsequences of ND steps are composed into a sequence of reasoning steps at some sort of *partial assertion* level. This sequence is then inserted in the proof graph as a potential substitute for the original assertion level step, which is marked as a *summary*.

An example for such an expansion and partial recomposition is shown in Figure 1, which exposes a crucial inference in the Steamroller proof in two levels of abstraction. Both variants show subproofs indirectly deriving the categorization of the fox (f) as a meat eater, that is, the fox f does not eat grain g, $\neg EATS(f, g)$.

When the derivation is carried out by a single assertion level step ((1) in Figure 1), this can be paraphrased by 'The wolf either eats grain or, in case the fox eats grain and is smaller than the wolf, the wolf eats the fox. Since the wolf does not eat grain, the wolf does not eat the fox, and the fox is smaller than the wolf, it follows that the fox does not eat grain'. Apparently, this is a very bad argumentation. Though the facts mentioned provide a complete account of the justifications underlying the required reasoning, the way how this works is completely obscure at first sight. However, this is not surprising, since the assertion level step underlying this reasoning is composed of several cognitively complex inference steps, as the expansion to the ND level ((2) in Figure 1) demonstrates. In the general case, this expansion would be followed by a recomposition encompassing cognitively simple deductive syllogisms, yielding a representation on the partial assertion level. Since there are only cognitively difficult inference steps in this instance, the representations on ND and partial assertion levels are identical. Through this expansion, the compound inference step is decomposed into three simpler ones, two disjunction eliminations with a modus tollens in between. The sequence of inference steps can be paraphrased by 'Since wolves do not eat grain, it follows that wolves like to eat all animals smaller than themselves that like to eat plants. Since wolves do not eat foxes, it follows that foxes do not eat grain or that they are not smaller than wolves. Since foxes are smaller than wolves, it follows that foxes do not eat grain.' With more skillful references to instantiations of the central axiom of this problem, this text can be improved to 'Since wolves do not eat grain, their eating habits imply that they are meat eaters. Since they do not eat foxes, it follows that foxes are not plant eaters or not smaller than wolves. Since foxes are smaller than wolves, foxes are not plant eaters, hence they are meat eaters' (see [14] for details on how these referring expressions are built).

4.2 Degrees of Explicitness

Unlike expanding summaries, creating omissions and short-cuts is driven by communicatively motivated *presentation rules*. They express aspects of human reasoning capabilities with regard to contextually motivated inferability of pieces of information on the basis of explicitly mentioned facts and relevant background knowledge [9]. These rules provide an interface to stored assumptions about the intended audience. They describe the following sorts of situations:

Cut-prop: omission of a *proposition* (premise) appearing as a reason
Cut-rule: omission of a *rule* (axiom instance) appearing as a method
Compactification: short-cut by omitting an *intermediate* inference step

These *reduction* rules aim at omitting parts of a justification that the audience is considered to be able to infer from the remaining justification components of the same line of the proof, or even at omitting an entire assertion level step that is considered inferable from the adjacent inference steps. In order for these rules to apply successfully, presentation preferences and conditions about the addressees' knowledge and inferential capabilities are checked.

The functionality of the reduction rules can be explained by a simple example. If trivial facts, such as $0 < 1$, or axioms assumed to be known to the audience, such as *transitivity*, appear in the set of justifications of some inference step, they are marked as *inferable* ($0 < 1$ through *Cut-prop*, and *transitivity* through *Cut-rule*, provided the use of an axiom is likely to appear evident from the instantiated form). Consequently, the derivation of $0 < a$ can simply be explained by $1 < a$ to an informed audience. Moreover, single facts appearing as the only non-inferable reason are candidates for being omitted through applying *Compactification*. If, for instance, $0 < a$ is the only non-inferable reason of $0 \neq a$, and $0 < a$, in turn, has only one non-inferable reason, $1 < a$, the coherence maintaining similarity between $0 < a$ and $1 < a$ permits omitting $0 < a$ in the argumentative chain. Altogether, $0 \neq a$ can be explained concisely by $1 < a$ to an informed audience.

For problems such as the Steamroller, which make reference to (pseudo-)real world knowledge, similar expectation-based omissions and short-cuts occur. For example, mentioning the size relation between two animals as an argument can be omitted, as in 'It follows that foxes are not plant eaters or not smaller than wolves. Hence, foxes are not plant eaters.' (an instance of a *Cut-prop*).

$$(1)\quad \frac{(EATS(w,g) \vee (EATS(f,g) \wedge (f < w)) \Rightarrow EATS(w,f)) \quad \neg EATS(w,g) \quad \neg EATS(w,f) f < w}{\neg EATS(f,g)}\text{Assertion}$$

$$(2)\quad \frac{\dfrac{\dfrac{\dfrac{(EATS(w,g) \vee (EATS(f,g) \wedge (f < w)) \Rightarrow EATS(w,f)) \quad \neg EATS(w,g)}{(EATS(f,g) \wedge (f < w)) \Rightarrow EATS(w,f)}\vee E \qquad \neg EATS(w,f)}{\neg EATS(f,g) \vee \neg(f < w)}\text{Modus Tollens}}{} \qquad (f < w)}{\neg EATS(f,g)}\vee E$$

Figure 1. An involved assertion level inference at two different levels of abstraction.

Let us look into more detail on how the inferential capabilities and assumptions about the background knowledge are expressed. Modeling these mental capabilities is done by distinguishing the following sorts of knowledge and communicative competence:

- knowledge per se, comprising (static) domain knowledge and (dynamic) referential knowledge,
- the attentional state of the addressee, determined by the pieces of knowledge in the current focus of attention,
- inferential skills, which comprise abilities to draw taxonomic, logical, and communicatively adequate inferences. The last kind of inferences concerns the capability to augment logically incomplete pieces of information in a given context.

The first component as well as taxonomic inferences are fairly standard, while logical inferences are a novel part in our model. Its operationalization, however, needs to reflect particularities of the domain. In our application, we use some simple stereotypes to express assumptions about the addressee's domain knowledge (see [6]). Domain knowledge is composed of the addressee's acquaintance with mathematical theories in terms of axioms, definitions, and associated hierarchical relations, while referential knowledge is incrementally built from the assertions made in the course of a proof presentation. For example, if a proof makes reference to a mathematical group, a competent addressee is immediately aware that there are unit and inverse elements in this group because they belong to the definition of groups, and he/she also knows the associated definitions. Moreover, if the proof mentions a subgroup, the addressee is also aware of the fact that the properties of ordinary groups apply to it. Consequently, proof presentation can directly make reference to these propositions without mentioning explicitly the underlying connections that are entailed in the explicit content representation. Thus, taxonomic inferences comprise the following kinds of reasoning:

- Propagating properties of mathematical objects along hierarchical relations.
- Expanding componential properties of mathematical objects.

The remaining components of our model, awareness and logical inferences, are expressed by the predicates AWARE-OF, COHERENT, and ABLE-INFER which are given domain-specific interpretations, elaborated for the domain of mathematics (formal details are given in [12]). For assessing the addressee's awareness (AWARE-OF), we test whether a piece of knowledge required is entailed in a list of theorems, definitions, and hierarchical relations assumed to be known to the addressee, which is expressed in a user model as simple stereotypes (see [6]). The underlying simplifying assumption is that being acquainted with some piece of generic knowledge is sufficient to be aware of it in the course of the entire proof. Inferential capabilities (ABLE-INFER) express whether a user is able to infer the missing pieces of knowledge to justify some conclusion, given only a subset of the premises. This reasoning process is approximated by the requirements that (1) composing the information given is sufficient to fully instantiate the entire inference step, and (2) matching the instantiated form with the relevant generic piece of knowledge is within the complexity limitations the addressee is assumed to be able to handle. The following inferential skills are distinguished, with limitations on the complexity of their applications:

- Generalizations of natural categories and instantiations of basic everyday knowledge; pieces of this sort of knowledge are represented as axioms in mathematical problems.
- And-eliminations to obtain an element on top level of a conjunction.
- Applications of modus ponens without any additional equivalence operations.
- Substitutions in axioms with constants or variables and at most one additional operator (such as a factor, or an exponent) replacing corresponding variables in generic expressions.
- Chaining inference steps with structurally identical conclusions, which differ only by constants or operators (operators must be related, such as '=' and '<').

The first three inferential skills are attributed to every user, the remaining ones only to users with some experience in mathematics.

A further issue to consider is the composition of such inference steps, which reflects the concept of coherence. According to psychological experiments, leaving out intermediate steps in a chain of argumentation should still be understood as a "direct" cause, while "indirect" causes negatively affect the reasoning effort [23]. In a previous approach to expert system explanations, this aspect has been modeled by requiring purposes of domain rules involved to be identical [11]. For proofs, we try to capture this coherence requirement by a structural similarity between intermediate and final conclusions: they must be joined by instantiation, generalization, part, or abstraction relations. Precise definitions for a larger set of operators and validation by associated empirical tests are still to be carried out. However, mentally inserting the missing pieces of information into a condensed representation in these sorts of situation is not without limitations. For example, the number of elements in a conjoined expression and its given presentation certainly influence the effort to pick a specific element, and the complexity of the substitution needed to obtain a required instantiation of some axiom or parts of an axiom may make this inference difficult. Hence, understanding the relation between expressions that are transducible into one another by the subsequent application of a substitution and several equivalence operations requires the exposition of some intermediate steps. For an extensive study examining the consequences of human memory limitations on the suitability of discourse contributions, see [24].

Applying the presentation rules to optimize the entire proof graph from an argumentative perspective is carried out in two processing cycles. In each cycle, the proof graph is traversed by starting from its leaf nodes and successively continuing to the root node, without back-tracking (that is, some sort of inverse depth-first search is invoked): In cycle one, *Cut-prop* and *Cut-rule* apply, marking locally inferable justifications. In cycle two, *Compactification* is invoked, adding alternative justifications through short-cuts, on the basis of the inferables. This order takes into account dependencies among the rules. It is also reasonably efficient, since only short-cuts require processing alternative lines of reasoning.

4.3 Interactive Exploration

In order to convey the information specified completely in view of the assumptions made about the audience, summaries are avoided and inferables are omitted. Depending on the target item, giving such an explanation in all details required for full understanding may result in a long text. Therefore, it is better to present a reduced first-shot contribution, which can be further investigated interactively, according to user reactions.

The associated communication is handled by *P.rex* by allowing three types of user interaction: A *command* tells the system to fulfill a certain task, such as explaining a proof. An *interruption* interrupts the system to inform it that an explanation is not satisfactory or that the user wants to insert a different task. In clarification dialogs, finally, the user is prompted to give *answers* to questions that *P.rex* asks when it cannot identify a unique task to fulfill. In this paper, we concentrate on interruptions.

The user can interrupt *P.rex* anytime to enter a new command or to complain about the current explanation. The following speech acts are examples for messages that can be used to interrupt the system:

`(too-detailed :Conclusion` C`)`
The explanation of the step leading to C is too detailed, that is, the step should be explained at a more abstract level.

`(too-abstract :Conclusion` C`)`
The explanation of the step leading to C is too abstract, that is, the step should be explained in more detail.

`(too-implicit :Conclusion` C`)`
The explanation of the step leading to C is too implicit, that is, the step should be explained more explicitly.

`(too-difficult :Conclusion` C`)`
The explanation of the step leading to C is too difficult.

(1) ***P.rex*:** Hence, $a \in U \cup V$ by the $\cup$-Lemma.

(2) User: This step is too difficult.

(3) ***P.rex*:** Since $a \in U$ or $a \in V$, $a \in U \cup V$ by the $\cup$-Lemma.
Do you understand this step?

(4) User: No.

(5) ***P.rex*:** Do you understand the premises?

(6) User: Yes.

(7) ***P.rex*:** In order to prove that $a \in U \cup V$ let us consider the following cases.
Case 1: Let $a \in U$. That leads to $a \in U \cup V$ by the definition of $\cup$.
Case 2: Let $a \in V$. Then $a \in U \cup V$ by the definition of $\cup$.
Do you understand this step?

(8) User: Yes.

Figure 2. A clarification dialog in *P.rex*.

In *P.rex*, `too-difficult` is considered as an underspecified interruption. Figure 2 gives an example: When the user complains that the derivation of the conclusion in utterance (1) was too difficult, the dialog planner enters a clarification dialog to find out which part of the explanation failed to remedy this failure. During the clarification dialog, the system tries to distill whether the user failed to follow some implicit references (utterance (3)), whether one of the premises is unclear (utterance (5)), or whether the explanation was too abstract (utterance (7)). The control of the behavior of the dialog planner is displayed in Figure 3.

When generating a first-shot description, all possible reductions amount to relaxing the degree of completeness in which the information is presented. Four alternatives are examined, in ascending order of increasing information reduction:

1. Omitting the way how a piece of knowledge (a domain regularity) is applied.
2. Omitting that piece of knowledge.
3. Omitting premises of the inference (eventually, only some of them).
4. Omitting intermediate inference steps.

The choice among these options is based on assumptions about the audience and on the resulting balance of textual descriptions. In [13] we have defined and motivated some strategies for that, examples will be given in the next section.

When one or several intermediate inference steps are omitted (option 4 in the above list of items), some sort of ad-hoc abstraction is carried out. The sequence of enclosing inferences is abstracted into a set of propositions consisting of its conclusion and its premises, while the method how the conclusion is obtained, that is, the underlying sequence of inferences, is omitted. If there is evidence that some of the premises are more important or of more interest to the audience than the remaining ones, larger sets of premises can be reduced to sub-

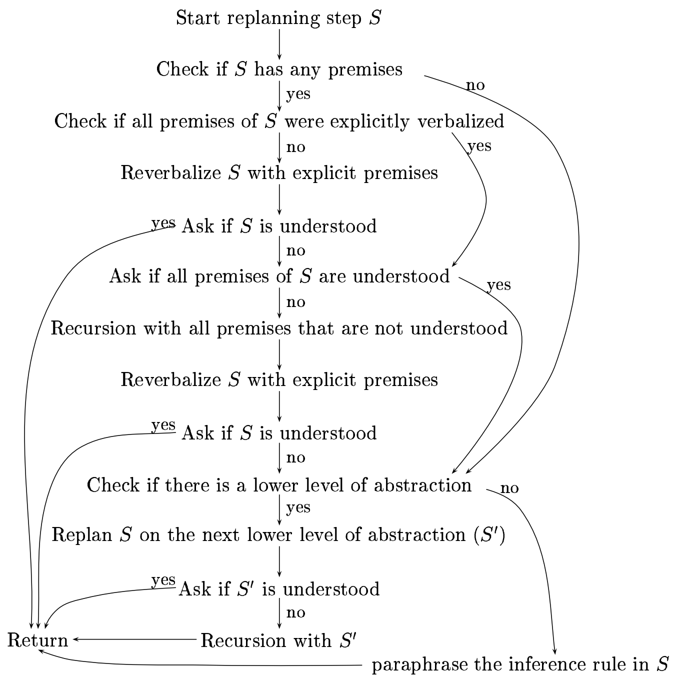

Figure 3. The reaction of the dialog planner if a step S was too difficult.

sets of these. In particular, this measure comprises preferring summaries over detailed exposition of involved inference steps. Moreover, in case these inferences constitute the expansion of a pre-designed proof method [19], which underlies the construction of a partial proof, the functionality of that method can be expressed by a descriptive phrase.

5 Explaining the Steamroller Proof

In this section, we demonstrate two strategies of building one-shot presentations of the solution to the Steamroller problem. In the examples, we paraphrase the expected output focusing on the structure and content of the produced text. Apparently, the proof sketch given when introducing the Steamroller is far from being a complete and fully comprehensible explanation of the proof, since many details that are necessary to understand how the central axiom is applied in each case are not mentioned. On the other hand, a full exploration of the proof is inappropriate for interactive environments because of its length.

A full description of the proof (see Figure 4) is produced by introducing a basic structure according to the main proof steps. These proofs steps, which are easily recognizable in the underlying proof graph, are routed in the application of those domain rules, which are not part of the addressee's background knowledge. In our example, only the rule about the food of animals is considered to be of this kind, in contrast to rules about categories ('a fox is an animal') and size relations ('birds are smaller than foxes').

The task of the presentation module is then to suitably mediate between such a concise proof sketch and a fully expanded proof description. One option, reducing the *quality*, leads to the text in Figure 5, achieves a compromise by fully explaining only the derivation of the first key assertion (foxes do not eat grain), while it merely states the other two key assertions derived. Since all three key assertions are derived by the same rule, this information can be stated compactly, preceding the derivation descriptions. The resulting description aims at reducing the set of propositions to be conveyed by explaining only a part of the proof in detail. This is done by selecting the propositions omitted in such a way that they are maximally connected, to minimize the number of potential clarification questions, which might address the derivations of one of the two key assertions, but not any more specific detail.

The other possibility is reducing the *convenience*, which leads to the text in Figure 6. It achieves a compromise by providing details about all key assertion derivations. The reduction here is obtained by merely stating the key assertions derived in connection with the underlying facts without elaborating how the responsible rule is applied. As in the previous case, that rule is only mentioned once, preceding the

---

It is first derived that foxes do not eat grain. This ultimately follows from the assumptions that wolves do not eat grain and foxes are smaller than wolves, because animals who do not eat plants eat plant eaters smaller than themselves. Thus, either foxes do not eat grain or they are not smaller than wolves. Hence, only the first alternative is valid. Moreover, it is derived that birds eat grain because animals eat plants if they do not eat plant eaters smaller than themselves. Birds do not eat plant eaters because it is assumed that they do not eat snails, but snails are smaller than birds and they eat plants. Finally, it is derived that foxes eat birds, because animals either eat plants, which foxes don't do, or they eat plant eaters smaller than themselves. Birds are such plant eaters, and they are smaller than foxes. Since foxes eat birds, an animal is known that eats a grain-eating animal, q.e.d.

Figure 4. Fully-detailed presentation of the proof of Schubert's Steamroller.

The proof runs through applying three times the rule that animals either eat plants or all plant eaters smaller than themselves. It is first derived that foxes do not eat grain. This ultimately follows from the assumptions that wolves do not eat grain and foxes are smaller than wolves, because animals who do not eat plants eat plant eaters smaller than themselves. Thus, either foxes do not eat grain or they are not smaller than wolves. Hence, only the first alternative is valid. Similarly, it is derived that birds eat grain, and finally, that foxes eat birds. Since foxes eat birds, an animal is known that eats a grain-eating animal, q. e. d.

Figure 5. Quality-reduced presentation of the proof of Schubert's Steamroller.

The proof runs through applying three times the rule that animals either eat plants or all plant eaters smaller than themselves. It is first derived that foxes do not eat grain. This ultimately follows from the assumptions that wolves do not eat grain and foxes are smaller than wolves. Thus, either foxes do not eat grain or they are not smaller than wolves. Hence, only the first alternative is valid. Moreover, it is derived that birds eat grain. Birds do not eat plant eaters because it is assumed that they do not eat snails, but snails are smaller than birds and they eat plants. Finally, it is derived that foxes eat birds, because they are plant eaters, and smaller than foxes. Since foxes eat birds, an animal is known that eats a grain-eating animal, q. e. d.

Figure 6. Convenience-reduced presentation of the proof of Schubert's Steamroller.

exposition of further details. A potential justification for this presentation lies in augmenting the assumptions about the addressee's inferential capabilities – he/she is assumed to mentally apply a previously unknown recently mentioned rule to a number of facts.

The production of longer, but information-reduced, utterances can naturally serve the purpose of a summary meeting certain length parameters and content preferences. Moreover, these texts are well-suited as first-shot explanations in comparable discourse situations, based on known requirements or on tentatively made assumptions about the addressee. Further details may be exposed, guided by vague hints or by specific demands of the other conversant, who has at least the following options at his/her disposal:

- Assessments concerning choices made in building the condensed descriptions, such as 'be more concise' or 'be less concise', and 'emphasize why some intermediate conclusion holds', that is, elaborate on the underlying justifications, or 'emphasize how it is derived', that is, illustrate the underlying method.
- A general statement expressing that the description produced is considered insufficient in view of the explanatory demands.
- A request statement about extending specific portions of the description produced, such as 'Why does P(x) hold?' or 'How do Q(x1) to Q(xn) cause P(x)?'

The assessments are interpreted in terms of changes in the parameter setting that guide choices in producing descriptions. Moreover, they may be treated as indirect requests asking for modified or extended follow-up presentations, in case a presentation built according to these assessments provides additional information. Similarly, an unspecific request for clarification triggers the presentation of extensions to the previously given one, eventually by following an alternative strategy, in the spirit of reactive explanations, as demonstrated by Moore and Swartout [21]. Finally, when being confronted with a specific request, either a request for expanding the underlying method, or one for expanding the underlying justification, a dedicated description with the required information is produced. In all cases, communicating the additionally required material follows the same lines as the preceding first-shot dialog contribution. For complicated and argumentatively rich elaborations, similar constraints concerning the length of the utterance to be produced may prove to be relevant even for the chosen subset of information.

User: Why do birds eat grain?

System: Because animals eat all plants or all plant eaters smaller than themselves. Snails eat plants and are smaller than birds, and birds do not eat snails.

User: How is that derived?

System: For animals, it holds that (1) they eat all plants or (2) they eat all plant eaters smaller than themselves. (2) cannot hold for birds, because snails eat plants and they are smaller than birds, but it is known that birds do not eat snails. Hence (1) holds for birds.

User: Why do foxes and birds have their eating habits?

System: Foxes do not eat grain because wolves do not eat grain, so they eat plant eaters smaller than themselves. Since foxes are smaller than wolves, and it is assumed that wolves do not eat foxes, foxes cannot be plant eaters, because wolves eat plant eaters smaller than themselves. Birds eat grain because animals eat plants if they do not eat plant eaters smaller than themselves. Finally, foxes eat birds because animals either eat plants, which foxes don't do, since they do not eat grain, or they eat plant eaters smaller than themselves.

Figure 7. Follow-up questions to content-reduced presentation of the proof of Schubert's Steamroller.

Being confronted with a shortened explanation, the addressee can request more details selectively, according to his/her interests (see Figure 7). For instance, details about the derivation of one or more of the unexplained facts in the explanation following with reduced quality can be focused on by asking dedicated how-questions. Similarly, why-questions can be asked to learn about the rules underlying specific derivations in the explanation with reduced convenience.

6 A Potential Extension

In this section, we demonstrate that the material for deductive argumentation, when provided by machine-generated proofs, restricts the associated argumentation in its scope. We illustrate the kind of limitations and describe additional sources for argumentation, exemplified by a new perspective on the Steamroller proof.

6.1 An Inherent Limitation

Arguments about a proof as considered so far merely consist of two components:

- *What* is derived, that is, the claims, which are intermediate steps in a proof, and may serve as arguments for other derivations.

- *Why* some results has been derived, that is, the proper arguments, which are the justifications of a proof step.

In essence, the entire proof is made up of a sequence of arguments of this kind. It may be varied so that it is more detailed or more condensed, more implicit or more explicit, but it merely specifies the facts that make up a proof. Such a presentation is inherently limited in its communicative function - it supports a "passive" understanding, which is restricted to a *control* or *verification* perspective on a proof. As opposed to that, an essential task in deduction is not merely *understanding*, but actually *finding* a proof. This puts a *search* or *performance* perspective on a proof, an "active" understanding for which there are no clues in the proper proof presentation.

6.2 The Performance Perspective

In order to provide an argumentative basis for showing how the search for a proof is carried out, high-level strategic conceptualizations are essential driving forces. These conceptualizations must consist in a rather limited repertoire of fundamental and adaptive techniques, which are relevant for different kind of proofs, but with varying details in concrete uses. Hence, assuming the principled acquaintance with such a conceptualization, recognizing its applicability in a concrete case, and a skillful performance in actually applying it must be addressed in an argumentative conversation. This characterization is typical for human-oriented problem-solving, with a mixture of limited, but highly diverse pieces of knowledge and operational skills to combine them. It is in sharp contrast to the large-scale uniform knowledge representation and schematic reasoning, which is the typical process organization for machine-oriented purposes. Therefore, even high-level characterizations of a machine-found proof, such as the level of proof plans [20] constitutes an inappropriate level of description for human-oriented purposes - the plans are too many and each of them contains too many details to be meaningful to humans as memorizable conceptualizations.

For elementary mathematics and logics, which are the most realistic areas for being subject to tutorial purposes, there are only a few fundamental proof techniques. Among them are the partitioning into simpler subproblems and the transformation to a different representation/calculus which allows for operations for which the original representation is inappropriate. The latter concept, for example, may be applicable in various contexts, including a transformation of assertions about residue classes into integer equations, and a transformation of operations on sets into propositional logic expressions. For humans, it is essential to recognize the commonality between the measures in each of these contexts. For addressing the domain of limit theorems, a method called "complex-estimate" has been developed as part of a proof planning for this domain [19]. This method is a specific form of the fundamental concept "partitioning into simpler subproblems", with a specific interpretation suitable for polynomial expressions. Since the method does not separate the (general) underlying concept from the (domain-specific) interpretation, which would render its application in automated proof planning considerably more difficult, it does not provide an adequate basis for argumentation about human problem-solving.

6.3 An Example – the Steamroller Proof

In our running example, the general concept underlying the problem-solving process is the *reduction of alternatives.* The relevance of this concept becomes apparent from the relation between the theorem to be proved and the formulation of the major piece of knowledge introduced in the problem definition. While the former states an eating relation between two animals, the latter specifies alternative possibilities for the eating habits of animals. Once the stategic value of the crucial problem-solving concept is recognized, the question arises how it can be applied in the given case. Since the alternatives are directly encapsulated in a rule, it is advisable to simply instantiate this rule so that it becomes evident which of the alternatives is true and which is false for a concrete instantiation. However, actually performing the instantiation may impose difficulties on a person unexperienced in problem-solving, since there are five animals in the context and each of them is a candidate for instantiating the two slots in the domain rule in question. In order to avoid exhaustive searching whenever possible, another general problem-solving principle can be taken into account, namely "look for most plausible instantiations first". The key for establishing a preference among all candidates lies in the relation "much smaller than" which must hold between the two animals referred to in the central axiom. Through this relation, more than half of the possible combinations are excluded. It is even more than that since the relation "much smaller than" is interpreted in a non-transitive way in the Steamroller problem, otherwise there would be multiple solutions. Altogether, less than a handful of combinations remain, which can be tested in turn. Carrying out these tests amounts to applying ND inferences, which depends on the structure of instantiated subclauses. For rules of comparable complexity as the central axiom in the Steamroller problem, an unskilled person may require argumentative support as well. In particular, testing this rule with the instantiations 'fox' and 'bird' may turn out to be difficult – it is not known a priori whether or not the fox eats grain, hence, both eating options for the fox must be maintained at first. It is only the indirect clue from examining the eating habits of the wolf which clarifies the fact that the fox does not eat grain. Appropriate arguments about guiding the search in such a setting will certainly be profitable for a student.

Finally, we illustrate the functionality of this broader argumentation by a fragment of an interactive construction of the proof to the Steamroller problem, in a tutorial environment, exemplified by the hypothetical dialog in Figure 8.

The system statements in this conversation are, in fact, not proper argumentations, but some kind of hints. However, in order to produce these hints, relying on an argumentative basis as outlined above is absolutely necessary. Hence, putting these extensions to live is very likely to improve dialog capabilities in tutorial environments in an essential way.

7 Conclusion

The analysis of human proof explanations shows that certain logical inferences are only conveyed implicitly drawing on the discourse context and default expectations. Moreover, different syllogisms call for different presentation strategies to account for human performance. In this paper, we proposed

...

System: How do you propose to solve that problem?

User: I do not know.

System: In order to obtain more information, you should try to resolve undecided alternatives. Can you identify where they are?

User: The eating habits of the animals.

System: Exactly. Can you find out about some?

User: I should look at that longish rule, but I do not know where to start.

System: This refers to animals, so what animals?

User: Any pair may be relevant.

System: Look more closely at the rule about their eating habits.

User: One of them must be much smaller than the other.

System: Perfect. Can you name such a pair?

User: The fox and the bird.

...

Figure 8. Fragment of a tutorial dialog about finding a proof.

the partial assertion level as an appropriate representation of proofs to plan the content of an explanation and different degrees of explicitness and condensation. Then, driven by the unfolding dialog, a reactive planner allows for an interactive, user-adaptive navigation through the proofs.

So far we have implemented *P. rex* and some tools for mediating between levels of abstraction. We are currently investigating manipulations of the proof structure to realize different degrees of explicitness. We will soon incorporate this work into a newly starting project on dialog-oriented tutoring systems. Moreover, we believe that our approach also proves useful for argumentative dialog systems in general.

REFERENCES

[1] Peter B. Andrews. Transforming matings into natural deduction proofs. In *Proceedings of the 5th International Conference on Automated Deduction*, pages 281–292. Springer Verlag, 1980.

[2] Daniel Chester. The translation of formal proofs into English. *AI*, 7:178–216, 1976.

[3] Yann Coscoy, Gilles Kahn, and Laurent Théry. Extracting text from proofs. In Mariangiola Dezani-Ciancaglini and Gordon Plotkin, editors, *Typed Lambda Calculi and Applications*, number 902 in Lecture Notes in Computer Science, pages 109–123. Springer Verlag, 1995.

[4] Bernd Ingo Dahn, J. Gehne, Thomas Honigmann, and A. Wolf. Integration of automated and interactive theorem proving in ILF. In William McCune, editor, *Proceedings of the 14th Conference on Automated Deduction*, number 1249 in LNAI, pages 57–60, Townsville, Australia, 1997. Springer Verlag.

[5] Andrew Edgar and Francis Jeffry Pelletier. Natural language explanation of natural deduction proofs. In *Proceedings of the 1st Conference of the Pacific Association for Computational Linguistics*, Vancouver, Canada, 1993. Centre for Systems Science, Simon Fraser University.

[6] Detlef Fehrer and Helmut Horacek. Exploiting the Addressee's inferential capabilities in presenting mathematical proofs. In Martha E. Pollack, editor, *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 959–964, Nagoya, Japan, 1997. Morgan Kaufmann.

[7] Armin Fiedler. Dialog-driven adaptation of explanations of proofs. In Bernhard Nebel, editor, *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1296–1300, Seattle, WA, 2001. Morgan Kaufmann.

[8] Gerhard Gentzen. Untersuchungen über das logische Schließen I & II. *Mathematische Zeitschrift*, 39:176–210, 572–595, 1935.

[9] H. Grice. Logic and conversation. *Syntax and Semantics*, 3:43–58, 1975.

[10] Amanda M. Holland-Minkley, Regina Barzilay, and Robert L. Constable. Verbalization of high-level formal proofs. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99) and Eleventh Innovative Application of Artificial Intelligence Conference (IAAI-99)*, pages 277–284. AAAI Press, 1999.

[11] Helmut Horacek. A model for adapting explanations to the user's likely inferences. *User Modeling and User Adapted Interaction*, 7:1–55, 1997.

[12] Helmut Horacek. Presenting proofs in a human-oriented way. In Harald Ganzinger, editor, *Proceedings of the 16th Conference on Automated Deduction*, number 1632 in LNAI, pages 142–156, Trento, Italy, 1999. Springer Verlag.

[13] Helmut Horacek. Tailoring inference-rich descriptions through making compromises between conflicting cooperation principles. *Int. J. Human-Computer Studies*,53:1117–1146, 2000.

[14] Helmut Horacek. Expressing references to rules in proof presentations. In R. Goré, A. Leitsch, T. Nipkov, *International Joint Conference on Automated Reasoning*, Short papers, Technical Report DII 11/01, University of Siena, pages 76–85, 2001.

[15] Xiaorong Huang. Reconstructing proofs at the assertion level. In Alan Bundy, editor, *Proceedings of the 12th Conference on Automated Deduction*, number 814 in LNAI, pages 738–752, Nancy, France, 1994. Springer Verlag.

[16] Xiaorong Huang and Armin Fiedler. Proof verbalization as an application of NLG. In Martha E. Pollack, editor, *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 965–970, Nagoya, Japan, 1997. Morgan Kaufmann.

[17] Philip Johnson-Laird and Ruth Byrne. *Deduction*. Ablex Publishing, 1990.

[18] Christoph Lingenfelder. *Transformation and Structuring of Computer Generated Proofs*. PhD thesis, Universität Kaiserslautern, Kaiserslautern, Germany, 1990.

[19] Erica Melis. AI-techniques in proof planning. In Henri Prade, editor, *Proceedings of of the 13th European Conference on Artifical Intelligence*, pages 494–498, Brighton, UK, 1998. John Wiley & Sons, Chichester, UK.

[20] Erica Melis and Jörg Siekmann. Knowledge-Based Proof Planning. *Artificial Intelligence*, 115(1):65–105, 1999.

[21] Johanna Moore and Bill Swartout. A reactive approach to explanation. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1504–1510, Detroit, MI, USA, 1989. Morgan Kaufmann.

[22] Mark E. Stickel. Schubert's steamroller problem: Formulations and solutions. *Journal of Automated Reasoning*, 2:89–101, 1986.

[23] Manfred Thüring and Kurt Wender. Über kausale Inferenzen beim Lesen. *Sprache und Kognition*, 2:76–85, 1985.

[24] Marylin Walker. The effect of resource limitations and task complexity on collaborative planning in dialogue. *Artificial Intelligence*, 85:181–243, 1996.