Natural Deduction Assistant (NaDeA)

Villadsen, Jørgen; From, Andreas Halkjær; Schlichtkrull, Anders

Publication date: 2018

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):
Natural Deduction Assistant (NaDeA)

Jørgen Villadsen, Andreas Halkjær From, and Anders Schlichtkrull

DTU Compute, AlgoLoG, Technical University of Denmark, 2800 Kongens Lyngby, Denmark
jovi@dtu.dk s144442@student.dtu.dk andschl@dtu.dk

Abstract

We present the Natural Deduction Assistant (NaDeA) and discuss its advantages and disadvantages as a tool for teaching logic. In particular we compare our approach to natural deduction in the Isabelle proof assistant. NaDeA is available online:

https://nadea.compute.dtu.dk/

1 Introduction

The Natural Deduction Assistant (NaDeA) [5, 6] runs in a standard browser and is open source software (https://github.com/logic-tools/nadea). Our formalization in the Isabelle proof assistant [4] of the syntax, semantics and the inductive definition of the natural deduction proof system extends work by Berghofer [1] but with a much more detailed soundness proof that can be examined and tested by the students. The corresponding completeness proof is also available but it is of course quite demanding. NaDeA can be used with or without installing Isabelle and it is not necessary that the students have any knowledge about proof assistants [2].

2 A Sample Proof

We consider the following formula and its online proof here: https://nadea.compute.dtu.dk/

$$\exists x. A(x) \rightarrow (\forall x. A(x))$$

Click Cancel help, Load, Test 9 and Load shown proof in order to obtain the finished proof.

Figure 1 shows the start of the proof – more or less - where a natural deduction rule is to be chosen in the proof step 2.

Figure 2 shows the finished proof – one can always click on Undo to go back all the way to the very first proof step 1.

Figure 1: A Sample Proof – Start
3 Selected Features for Students

We briefly describe a number of NaDeA features for students:

- Figure 3 shows the Welcome window. The Help button brings up the help window with this welcome information and a number of so-called hints.

- Figure 4 shows the Tutorial window. It contains a getting started guide as well as a list of the natural deduction primitives.

- Figure 5 shows the Exercises window. Solutions to all exercises are provided and can be revealed step-by-step with hints.

- Figure 6 shows the special NaDeA, soundness and completeness, window. The so-called verification button allows the user to verify any finished proof in Isabelle.

- Figure 7 shows the major Isabelle Code window – entitled: Definition of natural deduction proof system – with the formalization in Isabelle.

- Figure 8 shows the minor Isabelle Code window – entitled: Definition of first-order logic syntax and semantics – with the formalization in Isabelle.

There are several other NaDeA features for students – for example, the ProofJudge system in NaDeA can manage student assignments in courses with teaching assistants.

NaDeA uses the automation of our verified declarative prover \cite{3} tool to give students feedback on the provability of their goals and subgoals. Overall both that prover and NaDeA are related to the IsaFoL (\url{https://bitbucket.org/isafol}) project which unites researchers in formalizing logic in Isabelle. Among the formalizations in the project are SAT-solving, first-order resolution, a paraconsistent logic, sequent calculi and more.
4 Conclusion

NaDeA has been used for teaching first-order logic to hundreds of computer science bachelor students. NaDeA has recently been used by a class of mainly PhD students at the 29th European Summer School in Logic, Language, and Information (ESSLLI), University of Toulouse, France, 17-28 July 2017 (https://www.irit.fr/esslli2017/courses/24.html).

Proof assistants such as Isabelle allow for many kinds of reasoning that go beyond natural deduction and their interfaces, of course, account for that. In NaDeA, on the other hand, there are no distractions – all buttons and texts in NaDeA have to do with natural deduction. The structured environment provided by NaDeA, based on clicking buttons instead of textual input, makes it possible for students to focus on the proof development process. Since NaDeA only allows input of well-formed formulas and application of applicable rules, the student does not have to worry about neither syntax or well-formedness errors possible for instance in Isabelle. Conversely, experienced users may feel slightly inhibited by the system as textually inputting a formula is often faster than using the mouse. Furthermore for very simple proofs, students may be able to find them by clicking blindly and without understanding what they are doing, since only the applicable rules are shown. This is not a problem for larger proofs.

As future work we consider developing more teaching materials and making further evaluations of NaDeA as a tool for teaching logic.

Acknowledgements

We thank Alexander Birch Jensen for collaboration on the initial development of NaDeA and we thank John Bruntse Larsen and Stefan Berghofer for fruitful discussions.

References

Natural Deduction Assistant (NaDeA)  
Villadsen, From, and Schlichtkrull

Welcome to NaDeA: A Natural Deduction Assistant with a Formalization in Isabelle

NaDeA runs in a standard browser - preferably in full screen - and is open source software - please find the source code and further information here: https://logic-tools.github.io/

Use Tabs in your browser as necessary.

The following three buttons are available in the main proof window (the Escape key can always be pressed to cancel and go to the main proof window).

- Load: The Load button brings up the load window which allows for simple import/export of proof times. Also a number of so-called tests can be loaded as explained in the help window (see Exercises).
- Code: The Code button brings up the code window with the Isabelle formalization code for natural deduction as explained in the help window (see Tutorial). Also the complete Isabelle theory file with comments is available.
- Help: The Help button brings up the help window with this welcome information and a number of so-called facts. After this welcome information then please click on Tutorial and thereafter continue with Exercises...

In order to edit a proof, edit mode must be turned on (by default the edit mode is turned off). Edit mode can be toggled by clicking anywhere in the main proof window below the header.

In order to undo a step, click the Undo button (or Delete key). All previous proof states can be reached.

By clicking the Stop button (or Insert key), the undo sequence (performed up until Stop) becomes available for undoing such that the steps undone can be reached once more.

To the left of the Stop button there are two numbers "xy":
- x marks the position of the current state on the Undo stack. y is the total number of states on the stack which can all be reached by consecutive clicks on Undo.

The verification button represents the number of 'x' symbols in the current proof state (the number 1 in the above button). If (and only if) this becomes 0, the proof is finished.

When this is the case, the button may be clicked to verify the proof in Isabelle. It may also be clicked at any other time to read about the completeness of NaDeA.

Please provide feedback to: Associate Professor Jørgen Villadsen, DTU Compute, Denmark: https://people.compute.dtu.dk/jovi/

Figure 3: Welcome

Getting Started
1. To build the sample formula A → (A & B) → C, turn on edit mode and clicking in the square brackets. [] denotes the current list of assumptions which is initially empty. Use a predicate with no arguments.
2. After building the sample formula, apply the rule Int (introduction rules) to prove the formula A by assumption of A. The rule is also selected by clicking in.
3. The proof finishes automatically by applying the rule Assume since the formula A is found in the list of assumptions. It is finished because there is no pending "p".

Proof Judge
There are two components to Proof Judge:
1. The first component judges proofs while they are being developed. If a logical cannot be proved, its line number will turn orange like this . You can try this out by trying to prove P & Q and selecting a second for the goal to be checked.
2. The second component requires evaluation with a course and logging is by clicking the Proof Judge button shown above. Exercises in the course are available there and proofs can be handed in as answers to these. Furthermore proofs can be named and saved as drafts for later work.

Figure 4: Tutorial
Figure 5: Exercises

Figure 6: NaDeA, soundness and completeness
Figure 7: Definition of natural deduction proof system

Figure 8: Definition of first-order logic syntax and semantics