Towards a Playground for Logicians

Giselle Reis
Carnegie Mellon University
Doha, Qatar
giselle@cmu.edu

Abstract

Proving properties about proof systems is a necessary task to make sure that they are sound and satisfy basic criteria for the application being considered. Nevertheless, it is a tedious and error-prone task. In this talk, I will present ongoing work on the development of a trustworthy system that helps with the verification of proof calculi's properties. This system allows for a natural specification of sequent-style calculi and visualization of rules in LaTeX. The user is also able to construct simple proof trees. Moreover, it can be used to check identity expansion, cut-elimination and permutation lemmas with the click of a button. Our goal is to free researchers from having to write down big and repetitive proofs, and move towards the use of mechanized proofs when meta-theory of proof systems is concerned.

Keywords Proof theory, Mechanized proofs, Meta-theory

ACM Reference Format:
Giselle Reis. 2018. Towards a Playground for Logicians. In *Proceedings of Women in Logic Workshop (WiL'18).* ACM, New York, NY, USA, 2 pages. https://doi.org/

Proofs of properties about deduction calculi tend to follow the same pattern, with only a couple of more involved or exceptional cases. Moreover, the number of cases is combinatorial on the number of rules. Nevertheless, to be sure, one needs to develop them all. Currently, to the best of our knowledge, logicians follow one of two ways to perform such proofs: (1) pen and paper; or (2) proof assistant. Using a proof assistant has become a recent trend, but the learning curve and implementation details still present a big challenge. The amount of effort that goes into such automated proofs is so great that these are still considered scientific works of their own (e.g. [2]). If one needs to show cut-elimination of a proof system as a step towards another goal, the "quickest" is to use pen and paper.

Building on previous work, our goal is to develop an environment where logicians can easily specify and "play" with proof systems. Checking meta-theoretical properties will be a matter of clicking a button, and editing rules or building proofs can be done in a point-and-click interface. This way, they can quickly modify their rules of inference and see how they affect the properties and proofs. This system will be a combination of several independently developed parts.

WiL'18, July 08, 2018, Oxford, UK
2018.

SELLF Focused subexponential linear logic (SELLF) can be used as a framework for specifying and reasoning about proof systems [5]. In this framework, cut-elimination and identity expansion can be reduced to proving a formula in SELLF (using bounded proof search). Moreover, coupled with answer-set programming, we can use SELLF to decide when two inference rules permute [6]. These results were implemented in a system called (unsurprisingly) `sellf` [3].

Tatu Tatu [7] is a tool for automatically checking the admissibility of cut and non-atomic identity. The checks are sound, but not complete: cut-admissibility will only be concluded when a Gentzen-style cut-elimination proof is applicable.

Quati Quati [4] is a tool that can be used for checking permutability of inference rules. These are also sound but not complete. It has the added feature of displaying the rules of the object system in LaTeX, close to a way they would be written in a textbook. This is a way to sanity check the specification in the form of LL formulas.

Sequoia Sequoia [1] (under development) is a web application for inputting sequent calculus-like systems (in LaTeX) and building simple derivations in a point-and-click interface. The inference rules are translated into SML functions on the back-end that, when applied to an instantiated sequent, generate the necessary premises.

Putting it all together `sellf` is the back-end of Tatu and Quati, and thus proof calculi are specified as linear logic formulas in those systems. This is one of their biggest drawbacks. Sequoia lives independently and has a more natural interface for inputting calculi. Inference rules are internally represented as an SML datatype. We are currently working on an algorithm for translating objects of this datatype into `sellf` specifications. This way, the checks of Tatu and Quati can be used in a system where proof calculi is easily specified. We are also investigating how to use Quati's permutation lemmas in a more comprehensive cut-admissibility check.

Ultimately, we will have a system that can be used with minimal effort and provides some basic guarantees about proof calculi. The goal is to have an environment where logicians can quickly experiment with their systems before settling down on a specific design. In the future, we plan to export proof objects of the (possibly partial) proofs of meta-theoretical properties. This way they can be completed and checked independently (and formally).

Acknowledgments

The author would like to thank the following people who have greatly contributed to this project: AbdulRahman Alfayad, Leonardo Lima, Vivek Nigam, and Elaine Pimentel.

This publication was made possible by the generous support of the Qatar Foundation through Carnegie Mellon University in Qatar's Seed Research program. The statements made herein are solely the responsibility of the author.

References

[1] AbdulRahman Alfayad. 2018. Sequoia. (2018). https://github.com/AboudFayad/sequoia

[2] Kaustuv Chaudhuri, Leonardo Lima, and Giselle Reis. 2017. Formalized Meta-Theory of Sequent Calculi for Substructural Logics. *Electr. Notes Theor. Comput. Sci.* 332 (2017), 57–73. https://doi.org/10.1016/j.entcs.2017.04.005

[3] Leonardo Lima, Vivek Nigam, and Giselle Reis. 2011. `sellf`. (2011). https://github.com/meta-logic/sellf

[4] Leonardo Lima and Giselle Reis. 2016. Quati. (2016). http://quati.gisellereis.com/

[5] Vivek Nigam, Elaine Pimentel, and Giselle Reis. 2016. An extended framework for specifying and reasoning about proof systems. *J. Log. Comput.* 26, 2 (2016), 539–576. https://doi.org/10.1093/logcom/exu029

[6] Vivek Nigam, Giselle Reis, and Leonardo Lima. 2017. Automated Checking of Proof Transformations. (2017). Submitted to Journal of Automated Reasoning.

[7] Giselle Reis. 2012. Tatu. (2012). http://tatu.gisellereis.com/