Logic for Fun: demonstration

John Slaney
Australian National University

Abstract

Logic for Fun is an online tool aimed at teaching the skill of using first order logical notation to specify problems in such a way that a model of the encoding is a solution to the problem. Older versions of the tool have been in use for many years, but the latest (2026) offers increased functionality to improve both the user experience and the process of assessing students' work. In this demonstration, we give an overview of the site and illustrate the process of using it to solve problems.

2012 ACM Subject Classification Theory of computation Constraint and logic programming; Social and professional topics Computational thinking

Keywords and phrases Logic education, First order logic, Problem specification

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Category demonstration

1 The setting

Of all the skills we hope to inculcate in the students in our introductory logic classes, perhaps the most important is the ability to read and write—and understand—logical notation. The use of proof systems to advance mathematics, or automated reasoning systems to verify software is doubtless of intellectual and commercial importance, but the central rôle of logic across all of its applications is as a vehicle for representation. Not only is formalisation a prerequisite of any kind of logical reasoning, but it is of importance in its own right. The folk wisdom in the field of constraint programming, for instance, is that accurate problem specification is typically more than half of the solution. Moreover, while the ability to construct a natural deduction proof or an analytic tableau, or to remember what is meant by 't is free for x in φ' may be tolerably easy to teach and certainly easy to examine, for most of our students it is not practically significant once the semester is over. On the other hand, the ability to use logical concepts for analysis, clarification and disambiguation is the main skill they take away, so it should be a priority for logic education.

Logic for Fun was developed in response to a perceived need. Traditionally, we have attempted to train students in the use of logical notation by giving them examples—formulae to be rendered into natural language or natural sentences, often involving nested quantifiers or tangled connectives, to be expressed in symbols. It is common for students to find this kind of exercise both tedious and unrewarding; moreover, when they make mistakes, instead of studying our feedback and correcting their errors, they frequently dismiss it as over-pedantic and move on to something more interesting. At worst, they may disengage completely.

By contrast, many people enjoy doing logic puzzles as a pastime, to the degree that they are prepared to buy puzzle magazines and devote time to solving the simple logical problems they contain. The puzzles in such collections are almost always first order constraint satisfaction problems over (small) finite domains, so evidently these could function in place of the traditional formalisation exercises if students could be brought to work on them. Moreover, if there were an interactive piece of software providing the exercises, feedback on any formalisation errors would be immediate as well as invariably accurate. *Logic for Fun* was originally designed to do just that. The aim is to make the exercises less boring and to encourage engagement.

2 The tool

Logic for Fun addresses the need for students to practice the art of rendering natural language descriptions into formal logical notation. Rather than presenting them with example sentences to be translated, or with demands for formulae to be true or false of a given setup, it asks them to specify *problems*. There is a range of such problems on the site, starting with very simple "tutorial" examples introducing the notation and mechanics of the system and proceeding through logic puzzles requiring bijections between sets of five objects and the like, to more challenging problems taken from AI planning, diagnosis or combinatorics for which the user must invent appropriate ontology and vocabulary as well as using it accurately. The advantages of specifying problems rather than merely translating sentences are that the exercises are meaningful and that there is a sense of achievement when the black-box solver on the site reports a solution.

The logic of *Logic for Fun* is the classical first order one with truth-functional connectives and the standard quantifiers. In general, the process of logical analysis requires resources beyond those of plain first order logic: at least higher order constructions and an open-ended list of intensional operations are in the toolkit. However, what is basic is basic, so mastery of the art of logic begins with truth functions and standard quntification over domains of objects.

The logic used for fun is many-sorted rather than single-sorted; this removes the need to write tedious clauses ensuring for instance that a person, their job and a day of the week are different things. It also has built in support for (small) natural numbers, though not for operations on them beyond basic addition and subtraction, as the object is to learn logic, not mathematical programming. Predicate and function symbols may be declared and asserted to have certain useful properties such as "commutative" or "all different", again to obviate the need for boilerplate constraints. Partial funtions are allowed, though declared functions are total by default. Domains are assumed to be finite, as the background constraint solver looks for finite models only.

Logic for Fun has been through several versions over its 25-year history. Most importantly, it was completely re-scripted in 2014 and made free to the world. This year (2026) has seen another significant upgrade, with increased robustness and the addition of functionality that was missing from the 2014 version. Most visible to users is a facility giving them control over their saved work and any submissions they have made for the purposes of feedback or assessment. The back-end solver has also been improved, making the syntax for declarations more permissive, extending the possibilities for defining domains and adding a fully coherent semantic treatment of reference failures caused by partial functions.

3 The demonstration

The tool was described in a paper [1] delivered to the *Tools for Teaching Logic* conference in 2015, but no full demonstration of it has ever been given in such a forum. The present demonstration takes place in a series of steps:

1. The opening section of the demonstration covers creating an account, logging in, joining a group (such as an undergraduate class) and the basics of navigating the site.
2. We look at the form for writing a problem encoding and sending it to the solver, and then at the results this produces.
3. We step through encoding a simple problem, illustrating the many-sorted nature of the logic, the use of enumerated domains, the importance of quantifiers and other features.
4. We examine the list of problems provided on the site, noting some of their features.

5. We briefly show a tool for analysing sets of encodings by visualising a graph showing the syntactic "distance" between them. Not only does this reveal interesting differences of approach by separating clusters of encodings according to the structure they impose on the problem, but where the encodings are the submissions for an assignment, it acts as a fairly reliable plagiarism detector!
6. Finally, we run through an (anonymised) example of one student's use of the site over some hours, illustrating the processes which lead slowly towards understanding the logic of a problem.

References

1 John Slaney. Logic considered fun. In Antonia Huertas, João Marcos, María Manzano, Sophie Pinchinat, and François Schwarzentruber, editors, *Fourth International Conference on Tools for Teaching Logic, TTL 2015, Rennes, France, June 9-12, 2015. Proceedings*, pages 215–222. Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA), Université de Rennes 1, 2015.