# The AProS Project: Teaching Logic to Business and Engineering Students

Moris Polanco

Universidad Francisco Marroquín
Facultad de Ciencias Económicas
6a. calle final zona 10, Guatemala, Guatemala
mp@ufm.edu

**Abstract.** The paper discusses the use of the tools provided by the course *Logic & Proofs*, of the Open Learning Initiative (Carnegie Mellon University), for teaching Logic to freshmen students of Business Administration and Engineering at Francisco Marroquín University (Guatemala) over a period of two years. It is argued that the distinctive focus of the course—"strategic argumentation"—needs to be adapted to the students specific interests in order to be relevant to their education. Polya's "heuristic" approach to problem-solving is proposed as a complement to the course.

**Keywords:** Introduction to Logic, strategic thinking, automated proof search, Proof Lab, natural deduction, proof.

## 1 Context

The curriculum of Business Administration at Francisco Marroquín University (Guatemala) includes several humanities courses. Since 2009, Logic has been included as one, and is a required course for around one hundred and ninety students in their freshman year. Before the implementation of the L&P course in 2009, the students were required to take a course of Critical Thinking.

In 2009, a new program was opened in the School of Business Administration. It was specifically designed for students who would receive their degree (*licenciatura*) in Engineering, combined with Business Administration. Based on the profile of the students, it was decided to teach them Logic instead of Critical Thinking, the course taken by Business Administration (BA) students. Many options were explored in 2008 before the academic year started in January 2009, and finally the decision was made in favor of the course *Logic & Proofs* (L&P), of the Open Learning Initiative of Carnegie Mellon University. In 2010, the course of Critical Thinking for Business Administration students was replaced by a new course called "Logic and Critical Thinking," in order to reduce the gap between the two programs (BA and Engineering). While the Engineering students were taking Propositional and Predicate Logic, the BA students were taking Propositional Logic and Critical Thinking. For 2011, the plan is to teach the same course for both programs. Teachers will be required, however, to adapt the content to the interests and specific needs of their students.

P. Blackburn et al. (Eds.): TICTTL 2011, LNAI 6680, pp. 190–197, 2011.

## 2 Logic & Proofs

L&P is a fully web-based "introduction to modern symbolic logic. It provides a rigorous presentation of the syntax and semantics of sentential and predicate logic. The distinctive emphasis is on strategic argumentation."[1] It has been developed in the Laboratory for Symbolic and Educational Computability (LSEC) as part of the AProS—Automated Proof Search—Project, which "consists of four separate, but deeply integrated parts, namely, the central proof search engine Proof Generator, the Proof Tutor, the Proof Lab, and the web-based course Logic & Proofs." [2] That project is being directed by Wilfried Sieg.

In order to take the course, students are required to create an account at the Open Learning Initiative of Carnegie Mellon University and pay a fee.

The course consists of four parts: an Introduction (one chapter: Statements and Arguments), Sentential Logic (seven chapters: Syntax and Symbolization, Semantics, Derivations, Indirect Rules, Strategy and Derived Rules, Elementary Metamathematics), Predicate Logic (five chapters: Syntax and Semantics I, Syntax and Semantics II, Derivations, Strategies and Derived Rules, Identity and Functions), and Additional Topics, which is an excursion into Aristotelic Logic in one chapter. In our experience at UFM, the material can be covered in 14 to 16 weeks.

Like the rest of the courses of the Open Learning Initiative, *Logic & Proofs* includes a Grade Book and a Learning Dashboard. The Grade Book keeps track of the student's work and scores (which are given automatically) in each of the "learn by doing" sections, practice problems, practice lab problems, homeworks, labs, and exams. The instructors can even recreate any student's work, in order to see the steps he or she took to solve a particular problem. All the data can be exported to an Excel or Blackboard file, or printed out easily. In the Learning Dashboard, instructors get a complete report of how many students have been working on a particular module, how much of the module those students have worked, how much help students need in a module, and how individual students are doing in each module. This is very helpful for instructors. In fact, in my opinion, instructors must avoid the risk of putting too much attention on the scores, while losing sight of the learning objectives of the course.

L&P is complemented by a set of web-based tools that are indispensable in its design as a fully web-based introduction to modern symbolic logic. These tools are the Proof Lab, the Truth Lab, and the Proof Tutor, along with a series of small interactive learning environments along the lessons (called "Learn by Doing").

The Proof Lab, as its creators tell us,

> is a proof construction and student management system. The Lab offers a unique "backward-forward" proof representation through Fitch Diagrams. Students can apply rules backward in the Goal Tree, and forward in the Fitch Diagram. This representation allows students to employ in a very direct way the special strategies developed for the AProS Proof Engine.

> The Proof Lab handles "bureaucratic work" for students, allowing them to focus on the structure of arguments. On the one hand, it keeps track of justifications and applies rules in the Fitch Diagram, after they have been specified by the student. On the other hand, the Proof Lab points out mistakes, when students misapply a rule (for instance, applying elimination rule for the conditional to a conjunction) or mix separate subproofs. It provides the student with information to correct mistakes [3][1].

One of the advantages of the Proof Lab is that it minimizes the working memory load. Sweller [12] demonstrated that a high working memory load can interfere with problem solving.

The Truth Lab is the semantic counterpart of the Proof Lab. It was developed by Dawn McLaughlin, member of the staff of the AProS Project, in 2009. "As the Proof Lab engages the student in the conceptual essentials of proof construction, so the Truth Lab engages the student in the conceptual essentials and techniques that utilize the definition of truth in a fundamental way." [4] In 2010, the Truth Lab was greatly improved, and it now can be used to make truth tables along with truth trees.

Finally, the Proof Tutor, started in 2008,

> is the bridge between Proof Generator and the Proof Lab. It enables students who are stuck on a proof to receive hints, dynamically obtained from Generated Proofs. If a student requests a hint, Proof Generator will construct a complete proof, which the tutor analyzes. The tutor then provides hints to construct the proof using the efficient and natural strategies employed by Proof Generator. The first hint provided at any point in the proof is a general strategic one, and subsequent hints provide more concrete advice as to how to proceed. The last hint in the sequence recommends that the student take a particular step in the proof construction [2].

According to Douglas Perkins, "the goal of the proof tutor is to provide high level advice for students on proofs with the intent that students will learn to incorporate techniques from hints into their own reasoning." [6] The convenience of providing immediate feedback on errors, however, is a matter subject to discussion. Some argue that "the chief benefit of immediate feedback is to reduce time learning substantially (Anderson, Koedinger and Pelletier, 1995). However, others point out that the immediate feedback can interfere with aspects of learning" [8].

In my opinion, the Proof Tutor follows in line with the "general consensus" that "seems to be emerging on the context of advice messages":

> When error feedback is presented, it should generally just signal the error without commenting. This enables the student maximum opportunity to analyze the correct situation. When advice is given, the most

[1] In the AProS home page (http://www.phil.cmu.edu/projects/apros/) there are four videos worth watching, about the course in itself, the Proof Lab, The Truth Lab, and the Tutor.

> cost-effective content focuses on "reteaching" a correct analysis of the situation rather than debugging misconceptions. This correct analysis should be administered in at least three or four stages: (1) a reminder of the problem-solving goal, (2) a description of relevant features of the current problem state and the desired goal state, (3) a description of the rule for moving from the current state to the desired state, and (4) a description of a concrete action to take [8].

What makes L&P more than a simple web-based course is that the developing team is working continually on its development. The surveys provided by hundreds of students from different backgrounds, as well as the instructors feedback, provide them with the information needed to make the necessary adjustments to the content or to the tools[2].

## 3 The Strategic Approach

The strategic approach is the distinctive emphasis of L&P. Basically, it is "an ordered sequence of tactical steps" [1] that leads the student to an efficient proof construction. The tactical steps that the student is required to take, in sequence, are Extraction, Conversion, Inversion, Division, and Refutation. To apply these steps, the student must have a clear notion of every one of them, which in turn requires an understanding of the notion of "positive subformula."

The authors of the course assure the student that

> If you follow this procedure to the letter you are guaranteed to successfully complete your derivation—eventually—that guarantee being given by virtue of the fact we mentioned at the beginning. . . At first, you might find it a bit tedious to work through the procedure step by step, but if you keep with it, eventually it will become second nature and you will find that you are able to complete derivations with a minimum of effort. [1]

In the 2010 edition of the course, the procedure is nicely illustrated by flowcharts, and is explained in detail as an algorithm. Besides that, the students have the help of the Proof Tutor for some practice problems, which guide them in the application of the algorithm.

In my experience teaching L&P for two years, the potential problems for the strategic approach are at least two:

---

[2] At the end of the Spring course of 2007, for instance, Douglas Perkins ended a presentation with the following remarks regarding the Proof Tutor: "Where to go from here? The tutors described here show a comprehensive way to use an automated theorem prover to dynamically produce hints for students in proof search. Just how good is this tutor? While it can provide useful hints to students, what will its eect be on the rst few weeks of [working] with proofs? What components of it are the most useful? These questions may be addressed this winter, using logging data from the fall term." [13]

First, many students see the tactics as an unnecessary procedure for solving problems. “Much of the time, students look at the section of a partial proof and see immediately how to finish it.” [6] In other cases, they proceed by guessing, and do not care if they take unnecessary steps. The Proof Lab, in this respect, can be a dangerous tool, since the students may arrive at a proof applying rules indiscriminately, taking advantage of the “memory” of the Proof Lab. In this respect, it is of crucial importance that the students learn also to prove arguments using just pen and paper, and compare their procedure with the one they follow using the Proof Lab[3].

Second, the strategic approach can be “too mechanical.” According to Gilmore, the “emphasis is not on how well the user achieves the current task goals, but on how well they learn about the nature of that task in some general, abstract way.” [7] Perkins acknowledges Gilmore’s criticism, and points out that “successful tutoring ought to be responsive to the skills of the student, and this holds for strategic proof tutoring just as it does elsewhere,” and even recognizes that “it can be instructive . . . to allow students to use inference rules in valid but unproductive ways . . .” [6]

Some advantaged students may understand the benefits of the strategic approach, once they have also learned its *raison d’être*. Even recognizing that it is an algorithm, they must understand how their creators arrived at it. But that is not easy for the average student.

Douglas Perkins develops an interesting approach in his masters thesis for successful tutoring in propositional logic:

> To account for increasing skill as the student learns, then, there are three distinct tutoring levels or modes: tactical explanation, walking through a proof, and completing a partial proof. . . A tactical explanation is a goal-specific piece of information explaining which tactics can currently be employed, walking through a proof provides an example of how to think strategically, and completing a partial proof provides students with on-demands hints [6].

The main advantage of the “tactical explanation” is that it shows the student that “successful problem solving involves the decomposition of the initial problem state into subgoals and bringing domain knowledge to bear on those goals” [6].

The “walking through” level makes use of the Proof Tutor. Students are required to pay close attention to the hint it produces, and try to understand its justification.[4] The instructor must point out, though, that the hints make sense within the strategic approach that the Proof Lab uses.

[3] “As in any software environment, the efficiency with which the student can solve problems is an important consideration. However, the ultimate and unobservable product is knowledge. As a result, the assessment of educational software ultimately involves a transfer task: how well the students can solve problems working on their own outside the tutoring environment.”[8]

[4] The controversy about immediate feedback was outlined in the previous section.

The "completing proofs" level implies making a non-normal proof normal. "Normal proofs are preferred not because non-normal proofs are incorrect, but because non-normal proofs have extra clutter that is both cumbersome to the students cognitive load as well as simply unnecessary." [8] The student is guided by the instructor, with the help of the Proof Tutor, as to how to make a normal proof.[5]

## 4 Making Logic Relevant

Although the intrinsic value of Logic is quite obvious, teachers usually experience the need to "sell" the usefulness of Logic to students, particularly when they are not students of Philosophy or Mathematics. In the case of the three philosophers and one engineer that teach Logic to Business students at Francisco Marroquín, this is particularly critical.[6] Often, we are faced with the question "What is the use of it?" by students that are eager to know the best techniques to start a business, or how to perform well in marketing or finance. My usual answer to those students is that classes like Logic and Calculus will help them to take their thinking to more abstract levels, probably helping their brains to make more connections, and in that way enable them to be more creative and to "see" more business opportunities. But that is only a guess on my part, and I do not have empirical evidence to support it.

In a report presented in 1990, Wilfried Sieg (currently, the AproS Project Director) and Richard Scheines wrote that

> For our project [the Carnegie Mellon Proof Tutor] it was crucial to have a "theorem proving system" that can provide advice to a student user; indeed, pertinent advice at any point is an attempt to solve a proof construction problem. To be adequate for this task a system must be able to find proofs, if they exist, and to follow a strategy that in its broad direction is logically motivated, humanly understandable, and memorable. [9]

On the other hand, I share Corbertt, Koedinger and Andersons view that

> We are just entering a time when intelligent tutoring systems can have a real impact in the educational marketplace as technology costs decline. . . this will happen only if ITS [Intelligent Tutoring Systems] research focuses on *educational outcomes* as well as AI issues. . . [i]t is important for the field to remain focused on *valid pedagogical principles and educational outcomes* in exploring these areas. [8] (italics added)

[5] Perkins gives a detailed explanation on this topic on Appendix B of his thesis.

[6] In the case of the Engineering program taught at Francisco Marroquín, it is important to point out that it is not "pure engineering," but a mixture of Business Administration and Engineering. The degree that the graduates receive after a four-year study program is "Entrepreneurial Engineering."

In my opinion, we need to make a harder effort to adapt the ITS to valid pedagogical principles. In particular, we need to use our teaching methods to make clear the importance of proofs and sound argumentation. That way, students can better appreciate the value of the "theorem proving system" provided by the Proof Lab of the AproS Project. But besides that, I think that in order to be "humanly understandable, and memorable," the strategy that the system follows still needs to be tuned up.

In order to make the system more humanly understandable and memorable, my proposal is to incorporate Polyas "heuristic approach" [10] to problem solving to the "strategic approach" of L&P.

In *How to Solve it*, Polya [10] suggests the following steps when solving a mathematical problem (these steps are the summary of his "heuristic approach"):

> First, you have to understand the problem.
> After understanding, then make a plan.
> Carry out the plan.
> Look back on your work. How could it be better?

Of course, a logic problem is not a mathematical problem, and heuristic reasoning is not the same as making a formal proof. At the same time, it is important that the students understand the problem, not only in its logic form, but in its context. Examples taken from the LSAT, for instance, can illustrate the importance of logic reasoning for parliamentary debates. It would be a great improvement, in my opinion, if L&P contained more examples taken from everyday experiences, or from scientific problems.

It is true that Polya writes that it is bad "to mix up heuristic reasoning with rigorous proof" [10], since "[F]or a logician of a certain sort, only complete proofs exist. What intends to be a proof must leave no gaps, no loopholes, no uncertainty whatever, or else it is not a proof. Can we find complete proofs according to such a high standard in everyday life, or in legal procedure, or in physical science? Scarcely" [10].

However, Polya also says that "[W]e need heuristic reasoning when we construct a strict proof as we need scaffolding when we erect a building" [10], and that is why I think that heuristic reasoning can be a good complement to strategic thinking. Strategies and tactics only make sense in the context of the discovery.

We all want to avoid the situation described by Barwise and Etchemendy:

> Many students try to construct formal proofs by blindly piecing together a sequence of steps permitted by the introduction and elimination rules, a process no more related to reasoning than playing solitaire [11].

In order to do that, students need to understand the problem and make a plan, and I would argue that even before that, they need to see the problem, in its human or scientific context.

## 5 Conclusion

L&P is an excellent course for teaching Logic to university students. Its web-based tools are the state of the art in the field. Students can certainly master propositional and first-order logic in 14 to 16 weeks, provided that they are properly guided in their learning process. The web-based tools (the Proof Lab, the Truth Lab, and the Proof Tutor) need to be complemented with traditional teaching (blackboard and "pen and paper") to avoid becoming a device used by the students to deliver solved problems. "Transfer of knowledge to other environments, notably the non-tutor environment, is essential" [8].

On the other hand, I propose that the "heuristic approach" to problem solving as first presented by Polya in 1945 be incorporated to the strategic approach that is at the center of the L&P course. This goal requires the enrichment of the material presented in the course with problems and examples taken from the students everyday experiences, and from sciences that they are studying as part of their regular curriculum.

## References


1. Logic & Proofs, `http://oli.web.cmu.edu/openlearning/forstudents/freecourses/logic`
2. The AProS Project, `http://www.phil.cmu.edu/projects/apros/index.php`
3. Proof Lab, `http://www.phil.cmu.edu/projects/apros/index.php?page=prooflab`
4. The Truth Lab, `http://www.youtube.com/watch?v=UM1rCHH5wh8`
5. Logic& Proofs: The TruthLab, Proof Tutor, `http://www.phil.cmu.edu/projects/apros/`
6. Perkins, D.: Strategic Proof Tutoring in Logic. M. S. Thesis, Carnegie Mellon (2007)
7. Gilmore, D.J.: The Relevance of HCI Guidelines for Educational Interfaces, `http://nth.wpi.edu/classes/cs525t_tutoring_systems/paperforclass/gilmore-relevance%20of%20hci%20guidelines.htm`
8. Corbett, A., Koedinger, K.A., Anderson, J.: Intelligent Tutoring Systems. In: Helander, M., Landauer, T.K., Prabhu, P. (eds.) Handbook of Human-Computer Interaction, pp. 849–874. Elsevier Science B. V., Amsterdam (1997); Second, Completely Revised Edition
9. Sieg, W., Scheines, R.: Searching for Proofs (in Sentential Logic), Report CMU-PHIL-21 (1990)
10. Polya, G.: How to Solve It, 2nd edn. Princeton University Press, Princeton (1985)
11. Barwise, J., Etchemendy, J.: Logic, Proof and Language. CSLI Publications, Stanford (2008)
12. Sweller, J.: Cognitive load during problem solving: Effects on learning. Cognitive Science 12, 257–285 (1988)
13. Perkins, D.: Slides (pdf). Strategic Proof Tutoring in Logic, `http://dperkins.org/thesis/present.pdf`