Single Axioms in the Implicational Propositional Calculus

Frank Pfenning

Department of Computer Science
Carnegie Mellon University
Pittsburgh, Pa. 15213-3890
ARPANet: `fp@cs.cmu.edu`

Abstract

A class of challenge problems derived from a first-order encoding of the implicational propositional calculus is presented.

1 Introduction

In [1] (reprinted as [2]), Łukasiewicz presents the implicational propositional calculus and points out some single formulas which are complete as axioms. The implicational propositional calculus allows a straightforward encoding in first-order logic. This gives rise to some first-order theorems that seem to be very difficult to prove for human and machine alike. Łukasiewicz himself managed to prove the main theorem, namely that L_1 below is a single axiom.

I tested two of the best automated theorem provers on problems from the set. None were able to show that the Hypothetical Syllogism follows from the single axiom L_1, even when given considerable help with the insertion of lemmas into the initial set of clauses. The table in Section 4 summarizes my results.

This research was supported in part by the Office of Naval Research under contract N00014-84-K-0415 and in part by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 5404, monitored by the Office of Naval Research under the same contract.

To appear in the Proceedings of the 9th International Conference on Automated Deduction, Argonne, Illinois, May 23–26, 1988

2 Implicational Propositional Calculus

2.1 The Tarski-Bernays System

Tarski and Bernays described a system for reasoning in the implicational propositional calculus with three axioms and two inferences rules. The axioms are:

Simplification **(**S**):** $p \to (q \to p)$

Peirce's Law **(**P**):** $((p \to q) \to p) \to p$

Hypothetical Syllogism **(**H**):** $(p \to q) \to ((q \to r) \to (p \to r))$

The rules of inference:

Substitution: Substituting arbitrary formulas for propositional variables.

Detachment: From α and $\alpha \to \beta$ infer β for arbitrary formulas α and β.

2.2 Single Axioms

There are single axioms which allow one to derive all of the Tarski-Bernays axioms (and are hence complete). The problem dealt with in [1] is the question what the shortest such axiom would be. The author comes up with an axiom consisting of 13 characters, but also mentions other axioms with more characters, which were discovered earlier. Here are some:

L_1**:** $((p \to q) \to r) \to ((r \to p) \to (s \to p))$ (the shortest single axiom)

L_4**:** $((p \to q) \to (r \to s)) \to (t \to ((s \to p) \to (r \to p)))$

L_5**:** $((p \to q) \to (r \to s)) \to ((s \to p) \to (t \to (r \to p)))$

If we formulate the metatheory in first-order logic, formulas in implicational propositional calculus become terms, $\to$ becomes a binary function symbol, propositional variables become individual variables. We have a single monadic predicate Thm and a single meta-axiom formalizing the rule of *Detachment*:

$$D \equiv \forall p, q. \mathsf{Thm}(p) \land \mathsf{Thm}(p \to q) \supset \mathsf{Thm}(q)$$

The axiom of *Simplification* then becomes

$$S \equiv \forall p, q \ . \ \mathsf{Thm}(p \to (q \to p))$$

P and H are formalized similarly. In order to show that L_1 is a sufficiently strong single axiom, one has to derive

$$D \land \forall p, q, r, s \ \mathsf{Thm}(L_1) \supset S \land P \land H$$

The shortest known proof that L_1 implies H is given in the paper and consists of 29 applications of *Detachment*. Łukasiewicz notes (in 1947!):

> A formalized proof can be checked mechanically but cannot be mechanically discovered. I do not know of any method of finding proofs in the Implicational Propositional Calculus than the method of "trial and error."

3 Problem Summary

I briefly summarize the definitions of the problems, writing the binary function symbol "$\rightarrow$" in infix notation.

$$D \equiv \forall p, q \;.\; \mathsf{Thm}(p) \wedge \mathsf{Thm}(p \rightarrow q) \supset \mathsf{Thm}(q)$$

$$\begin{aligned}
L_1 &\equiv \forall p, q, r, s \;.\; \mathsf{Thm}(((p \rightarrow q) \rightarrow r) \rightarrow ((r \rightarrow p) \rightarrow (s \rightarrow p))) \\
L_4 &\equiv \forall p, q, r, s, t \;.\; \mathsf{Thm}(((p \rightarrow q) \rightarrow (r \rightarrow s)) \rightarrow (t \rightarrow ((s \rightarrow p) \rightarrow (r \rightarrow p)))) \\
L_5 &\equiv \forall p, q, r, s, t \;.\; \mathsf{Thm}(((p \rightarrow q) \rightarrow (r \rightarrow s)) \rightarrow ((s \rightarrow p) \rightarrow (t \rightarrow (r \rightarrow p))))
\end{aligned}$$

$$\begin{aligned}
I &\equiv \forall p \;.\; \mathsf{Thm}(p \rightarrow p) \\
S &\equiv \forall p, q \;.\; \mathsf{Thm}(p \rightarrow (q \rightarrow p)) \\
P &\equiv \forall p, q \;.\; \mathsf{Thm}(((p \rightarrow q) \rightarrow p) \rightarrow p) \\
H &\equiv \forall p, q, r \;.\; \mathsf{Thm}((p \rightarrow q) \rightarrow ((q \rightarrow r) \rightarrow (p \rightarrow r)))
\end{aligned}$$

D is always used as a clause, and one of the L_i is selected as additional assumption. Then one tries to prove one I, S, P, and H, perhaps using one or more of the others as lemmas.

4 Problem Status

The following is a table indicating the status of various problems that arise as outlined above. These results are not intended to give a measure of power for the theorem provers involved. Rather, they are meant to indicate the order of difficulty of the problems posed. Only the default heuristics were used.

Label	Theorem	Prover A[1]			Prover B[2]		
		Length	Time	Inferences	Length	Time	Inferences
I_1	$L_1 \supset I$	11	70	374	20	28	12,745
S_1	$L_1 \supset S$	11	90	395	14	1	592
P_1	$L_1 \supset P$		failed		30	—[3]	2,352,964
IP_1	$L_1, I \supset P$	14	1585	10,383	18	—[3]	19,359
H_1	$L_1 \supset H$		failed			failed	
IH_1	$L_1, I \supset H$		failed			failed	
IPH_1	$L_1, I, P \supset H$		failed			failed	

The times are in cpu seconds on a Sun 3/260, "Length" is the length of the proof found, "Inferences" is the number of logical inferences done during proof search. "$\supset$" indicates which assumptions were used in addition to the clause D. The subscript to the label indicates which single axiom was used in the experiment (I only used L_1). "Failed" means that the theorem prover ran for several hours without finding a proof.

With respect to the still automatically unproven theorems, note that Łukasiewicz's proof has length 29, its longest formula has 31 characters, and the deepest nesting of implications is 5. Unfortunately we don't know how long he worked on the proof.

References

[1] Jan Łukasiewicz. The shortest axiom of the implicational calculus of propositions. *Proceedings of the Royal Irish Academy*, 52(3):25–33, April 1948.

[2] Jan Łukasiewicz. The shortest axiom of the implicational calculus of propositions. In L. Borowski, editor, *Jan Łukasiewicz, Selected Works*, pages 295–305, North-Holland, 1970.

[1] Release 0 of D. Plaisted's C Prolog theorem prover based on the simplified problem reduction format with default heuristics.

[2] Version 1e of M. Stickel's Prolog technology theorem prover in Sun Common Lisp with default heuristics.

[3] Comparable times not available, since run on different systems.