Hilbert's 24th Problem, Proof Simplification, and Automated Reasoning*

Larry Wos

Mathematics and Computer Science Division
Argonne National Laboratory
Argonne, IL 60439
wos@mcs.anl.gov

1. Perspective and Significance

Do you ever wonder about the relevance of our field, automated reasoning, to the interests of great mathematicians and logicians of decades ago? Which of them would evince far more than mere curiosity in what is occurring and what might occur with the assistance of a powerful automated reasoning program? Who, among them, would present questions for us to study with such a program? Would any of the very famous ones be indeed impressed with the reasoning power now available? Well, at least one mathematician of overwhelming note would find some of our research far more than pertinent. Here, in this notebook (in which I rely on the clause notation), I shall show that this observation is most accurate and provide much evidence of the interest he would be showing, if he were still with us.

Of course, you know to whom I am referring: Yes, Hilbert is that mathematician. As I show how and why Hilbert would be interested in automated reasoning, I shall (at least implicitly) offer you challenges, results that you might try to improve upon. If you wish to experience the excitement of working on a problem offered by Hilbert, in one of the contexts to be featured here, this notebook will offer various appropriate techniques; no algorithm is known to me.

If you have read any of my other notebooks, you are undoubtedly aware of my delight in having access to W. McCune's awesome program OTTER as a reasoning assistant. The power of this program is illustrated by its use in answering open questions (some of which had remained unanswered for many decades), settling conjectures, finding new single axioms, discovering new or more elegant proofs, and—an activity that may indeed be unfamiliar to many researchers—addressing in various ways the recently found Hilbert problem, his 24th. (I shall include various elegant proofs here, some drawn from areas of logic and based on the use of the inference rules condensed detachment and hyperresolution, and some taken from algebra based on the use of the inference rule paramodulation, which treats equality in a sophisticated manner.) Indeed, thanks to R. Thiele and his thorough study, in the late 1990s, of Hilbert's notebooks, we now know of a problem Hilbert posed.

Yes, I do mean his 24th. As discovered by Thiele and featured in the *American Mathematical Monthly* in early 2003 and in an article Thiele and I wrote for the *Journal of Automated Reasoning*, Hilbert was seriously interested in a problem that he did not present in his 1900 lecture in Paris, where he offered the famous twenty-three problems that have occupied the attention of researchers for decades. That twenty-fourth problem, which Hilbert noted that he did not have time to make precise, concerns *proof simplification*.

Specifically, in his notebook Hilbert wrote:

"The twenty-fourth problem in my Paris lecture was to be: Criteria of simplicity, or proof of the greatest simplicity of certain proofs. Develop a theory of the method of proof in mathematics in general. Under a given set of conditions there can be but one simplest proof. Quite generally, if there are two proofs for a

*This work was supported in part by the Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357.

theorem, you must keep going until you have derived each from the other, or until it becomes quite evident what variant conditions (and aids) have been used in the two proofs. Given two routes, it is not right to take either of these two or to look for a third; it is necessary to investigate the area lying between the two routes."

Although proof simplicity is far from a rigorous concept, the most obvious way to simplify a proof is to shorten it—more formally, to find a second proof with fewer deduced equations or formulas than are present in the proof in hand. (With OTTER and various methodologies, I have shortened many a proof, in one case from roughly 1,800 equations to roughly 200.) A quite different type of proof simplification is that in which the simpler proof avoids the use of thought-to-be-indispensable lemma. (In Lukasiewicz's infinite-valued sentential calculus, I found a proof that dispenses with the use of three key lemmas, numbered by Rose and Rosser 2.22, 3.5, and 3.51; see Section 4.3.) A still different type of simplification focuses on the variable richness of a proof, the maximum number of distinct variables that occurs among the deduced steps. And, as you correctly surmise, other types of proof simplification merit study, a topic for later in this notebook.

In this notebook, I highlight a few of the contributions I—and OTTER—have made in simplifying proofs and, especially for the person who might also study the Hilbert problem, I discuss some of the approaches that were effective. I also indicate the limited progress I have made on the second part of Hilbert's 24th problem: formulating a theory of simplicity.

2. Proof Shortening

The year was 1979, if memory serves, when the fine logician John Kalman made his world tour of institutions that focused on what was then called automated theorem proving. Early in his tour, he visited Argonne National Laboratory. When I met him there, he told me of seven formulas in equivalential calculus whose status, with regard to being a single axiom, was at the time unknown. One of these formulas was the following, *XHN*.

```
P(e(x,e(e(y,z),e(e(z,x),y)))).  % XHN
```

That area of logic is typically studied with the inference rule *condensed detachment*, captured with the following clause in the presence of hyperresolution, where "|" denotes logical **or** and "-" denotes logical **and**.

```
% Following clause is for condensed detachment.
-P(e(x,y)) | -P(x) | P(y).
```

The function symbol *e* denotes "equivalent", and the predicate P denotes "is provable". My colleague Steve Winker answered the open question focusing on *XHN*, showing that it is in fact a single axiom. His proof has length 159 (steps of condensed detachment).

At that time, late 1970s and early 1980s, I was not concerned with proof shortening. That area became an interest of mine in the early 1990s, if I recall my history. Proof shortening continues, as is clear from my notebooks, to fascinate me. How pleased—and yes, astounded—I was to find, from Thiele, that proof shortening is so relevant to Hilbert's 24th problem. The discussion of this aspect of mathematics and logic will show how far the field of automated reasoning has advanced and even indicate that advances are occurring more rapidly in the past couple of years. With regard to *XHN*, in 2001 I with McCune's OTTER found a 19-step proof establishing that formula to be a single axiom. M. Stickel subsequently found an 18-step proof, in December 2004, by deducing the formula known as *UM*, which itself is a known single axiom, the following in negated form.

```
-P(e(e(e(a,b),c),e(b,e(c,a)))) | $ANSWER(P4_UM).
```

Even though roughly twenty-five years had elapsed from the time Winker succeeded, I hazard that Hilbert would have been quite satisfied with the reduction from 159 to 18. Of course, you can use as a target any known axiom system; one of the favorite targets consists of what is called symmetry and transitivity.

For a startling result (that I suspect Hilbert would have indeed enjoyed), in algebra (the theory of loops), in late 2004 I replaced a proof of length just short of 4,000 with one of length just under 2,000. (J.

D. Phillips was the source of the original proof.) The success was especially pleasing in that the former (and much longer) proof contains thousands of implicit steps (not counted among the nearly 4,000) through the use of automatic simplification and canonicalization (through the use of demodulation), whereas the latter contains none.

Decades after Hilbert wrote of the 24th in his notebooks, as the literature shows, proof shortening, or proof abridgment, was clearly of interest to fine logicians that include C. A. Meredith, A. Prior, I. Thomas, and D. Ulrich. (These logicians were undoubtedly unaware of the Hilbert problem.) In that regard, I offer yet another example that addresses the Hilbert 24th problem, one taken from the work of Meredith. He provided what may well be the shortest single axiom for two-valued sentential (or classical propositional) calculus, the following of length 21.

```
P(i(i(i(i(x,y),i(n(z),n(u))),z),v),i(i(v,x),i(u,x)))).
```

Meredith's proof, deriving a well-known 3-axiom system of Lukasiewicz, the following, has length 41 (steps deduced with condensed detachment).

```
% Luka 1 2 3.
P(i(i(x,y),i(i(y,z),i(x,z)))).
P(i(i(n(x),x),x)).
P(i(x,i(n(x),y))).
```

I believe it was in 1992 that I began seeking a shorter proof. Finally, in late 2000, I found a 38-step proof; Ulrich provided much of the impetus. (For the person who enjoys oddities, that 38-step proof deduces the most complex of the three Lukasiewicz axioms, the second, using it to complete the proof of all three.) You might find exceedingly challenging the goal of finding a proof of length 37 or less, if such exists.

If you wish to duplicate—or better, improve on—these successes and others I cite in this notebook, and (in effect) participate in attacking Hilbert's 24th problem in at least one aspect, you may wish to know how I proceed. But, perhaps disappointing, I can offer no algorithm for proof shortening or, for that matter, for finding a first proof. Each activity, from my experience, is an art, even with the aid of a program as powerful as McCune's OTTER is. Typically, I rely on a two-phase approach.

In the first phase, I take the steps of the proof in hand and instruct the program to key on them in preference to any other deduced equation or formula; one can use the resonance strategy or R. Veroff's hints strategy. In particular, with the *resonance strategy*, if a deduced conclusion matches (treating all variables as indistinguishable) one of the deduced steps of the proof in hand, it is given preference for initiating inference-rule application over any other deduced conclusion. The equations or formulas being keyed on (as resonators) are *not* treated as lemmas or as having a **true** or a **false** value. They are instead considered as attractive symbol patterns, each assigned a value reflecting a conjectured attractivity. In this phase, I typically employ McCune's *ancestor subsumption*, a procedure that compares derivation lengths to the same conclusion and, if such is found, prefers the strictly shorter. If a shorter proof is found, then the deduced steps of the new proof are keyed upon rather than those of the original proof. The process is iterative, continuing until no shorter proof (than the preceding) is discovered.

The second phase calls for the avoidance of steps occurring in a (best) proof that has been completed. Specifically, through means offered by OTTER, I invoke a procedure that considers the deduced steps of the proof under study (usually one at a time) and discards any newly deduced conclusion that is subsumed by the step in question. Since any step subsumes itself, the procedure blocks retention of the steps of the proof in hand, again usually one at a time. Sometimes, a given proof contains as a proper subproof (where parentage is ignored) a proof of the theorem of interest. (Consideration of the concepts of subproof and proof in the context of proof shortening can lead you to an interesting myth, to be discussed in Section 3.)

Yes, some of my experiments began with a proof A of length k (consisting of a set of hypotheses followed by k deduced steps) and led to the discovery of a proof B of length n with n (sometimes sharply) less than $k-1$ and—to my utter delight—with all n deduced steps among the k deduced steps of A. An option in this phase of proof shortening has one instruct the procedure to block (the use of) two steps at a time, or three, but seldom more than three; this phase is also iterative. A second option has the researcher instruct the program to rely in part on McCune's ratio strategy, a strategy that combines a breadth-first search with

that of complexity preference.

You can, when in search of a shorter proof, force the program to follow paths that perhaps have never been followed. The means to do so include placing a bound (smaller than that of the proof in hand) on the complexity of retained conclusions, limiting the number of distinct variables present in a retained conclusion, placing a higher emphasis on conclusions retained early in the search, and using a strategy known as *cramming*. Intuitively, in that strategy, the program is instructed to force, or cram, the deduced steps of a proper subproof into the proof of the theorem as a whole in a manner that such steps are used as parents twice, thrice, or more. An amusing side-effect of a successful use of the cramming strategy is that various proper subproofs of so-called intermediate steps are traded for longer subproofs of those steps; see the story at the end of Section 3. This phenomenon frequently occurs when proving a conjunction, where the focus is on individual members of the conjunction rather than on intermediate steps. This side-effect is acceptable because the goal is to find a shorter proof of the full theorem. (I shall have more to say about this later in the notebook.)

I often use OTTER to find shorter proofs at the end of studies having a quite different objective. I present here two examples, one involving axiomatic questions, the other searching for missing proofs.

2.1. Axiomatic Questions

One question that was open for approximately seven decades nicely captures what can be done with heavy reliance on an automated reasoning program. The question asks whether the formula *XCB* is sufficiently strong enough to serve as a complete axiom system for equivalential calculus.

$$XCB = e(x,e(e(e(x,y),e(z,y)),z))$$

Ulrich and I studied this question intensely. In the last stage of our study, applying various strategies and methodologies, after four days of real time and much CPU time, OTTER answered the question in the affirmative, deducing a known 2-basis consisting of symmetry and transitivity. The first proof showing *XCB* to indeed be a shortest single axiom has length 71 (applications of condensed detachment) and variable richness 12. (For the historian and for the researcher studying open questions, that 71-step proof was obtained in the presence of assigning the value 48 to max_weight, which in turn implicitly in this study assigns the value 12 to max_distinct_vars.) Fourteen shortest single axioms exist. *XCB* is the last of its kind.

The shortest proof I have found that deduces a known axiom system from *XCB* has length 22 (applications of condensed detachment), completing with the deduction of the cited 2-basis. (I am indebted to Stickel in that my discovery was based on his success, on December 24, 2004, with finding a 24-step proof; he used his program and, in effect, assigned the value 64 to max_weight, in contrast to the assignment of 48 used by Ulrich and me.) For the researcher who enjoys a substantial challenge, I suggest finding a shorter proof than length 22 (the following) that completes with the deduction of a known basis or proving that no shorter such proof exists.

A 22-Step Proof for XCB

```
----- Otter 3.3g-work, Jan 2005 -----
The process was started by wos on lemma.mcs.anl.gov,
Mon Apr  4 12:55:06 2005
The following has proofs of lengths  18.
The following has proofs of lengths  22.
-----> EMPTY CLAUSE at   0.15 sec ----> 1072 [hyper,91,847,428] $ANSWER(all_s_t_indep).

Length of proof is 22.  Level of proof is 17.

---------- PROOF ----------

90 [] -P(e(x,y))| -P(x)|P(y).
91 [] -P(e(e(a,b),e(b,a)))| -P(e(e(a,b),e(e(b,c),e(a,c))))|$ANSWER(all_s_t_indep).
```

93 [] P(e(x,e(e(e(x,y),e(z,y)),z))).
124 [hyper,90,93,93] P(e(e(e(e(x,e(e(e(x,y),e(z,y)),z)),u),e(v,u)),v)).
126 [hyper,90,124,93] P(e(e(e(e(x,e(e(e(x,y),e(z,y)),z)),u),v),e(u,v))).
128 [hyper,90,93,126] P(e(e(e(e(e(e(e(x,e(e(e(x,y),e(z,y)),z)),u),v),e(u,v)),w),e(v6,w)),v6)).
130 [hyper,90,126,93] P(e(x,e(e(e(e(e(y,e(e(e(y,z),e(u,z)),u)),x),v),e(w,v)),w))).
136 [hyper,90,126,130] P(e(x,e(e(e(e(e(y,e(e(e(y,z),e(u,z)),u)),e(e(v,e(e(e(v,w),e(v6,w)),v6)),x)),v7),e(v8,v7)),v8))).
142 [hyper,90,130,93] P(e(e(e(e(e(x,e(e(e(x,y),e(z,y)),z)),e(u,e(e(e(u,v),e(w,v)),w))),v6),e(v7,v6)),v7)).
149 [hyper,90,130,136] P(e(e(e(e(e(x,e(e(e(x,y),e(z,y)),z)),e(u,e(e(e(e(e(v,e(e(e(v,w),e(v6,w)),v6)),e(e(v7,e(e(e(v7,v8),e(v9,v8)),v9)),u)),v10),e(v11,v10)),v11))),v12),e(v13,v12)),v13)).
151 [hyper,90,126,136] P(e(x,e(e(e(e(e(y,e(e(e(y,z),e(u,z)),u)),e(e(v,e(e(e(v,w),e(v6,w)),v6)),e(e(v7,e(e(e(v7,v8),e(v9,v8)),v9)),x))),v10),e(v11,v10)),v11))).
174 [hyper,90,128,151] P(e(e(e(e(x,e(e(e(x,y),e(z,y)),z)),e(e(u,e(e(e(u,v),e(w,v)),w)),e(e(v6,e(e(e(v6,v7),e(v8,v7)),v8)),e(e(e(e(e(v9,e(e(e(v9,v10),e(v11,v10)),v11)),v12),v13),e(v12,v13)),v14)))),v15),e(v14,v15))).
204 [hyper,90,174,149] P(e(e(x,e(e(y,e(e(e(e(e(z,e(e(e(z,u),e(v,u)),v)),e(e(w,e(e(e(w,v6),e(v7,v6)),v7)),y)),v8),e(v9,v8)),v9)),x)),e(v10,e(e(e(v10,v11),e(v12,v11)),v12)))).
205 [hyper,90,174,142] P(e(e(x,e(e(y,e(e(e(y,z),e(u,z)),u)),x)),e(v,e(e(e(v,w),e(v6,w)),v6)))).
220 [hyper,90,93,205] P(e(e(e(e(e(x,e(e(y,e(e(e(y,z),e(u,z)),u)),x)),e(v,e(e(e(v,w),e(v6,w)),v6))),v7),e(v8,v7)),v8)).
226 [hyper,90,220,205] P(e(e(e(x,y),x),y)).
227 [hyper,90,220,204] P(e(e(e(x,e(y,e(e(e(y,z),e(u,z)),u))),v),e(x,v))).
284 [hyper,90,227,124] P(e(e(e(x,e(e(e(x,y),e(z,y)),z)),e(e(e(u,v),e(w,v)),w)),u)).
292 [hyper,90,126,284] P(e(e(e(e(x,y),e(z,y)),z),x)).
351 [hyper,90,93,292] P(e(e(e(e(e(e(e(x,y),e(z,y)),z),x),u),e(v,u)),v)).
383 [hyper,90,351,136] P(e(e(e(e(x,e(e(e(x,y),e(z,y)),z)),e(e(u,e(e(e(u,v),e(w,v)),w)),e(e(e(e(e(v6,v7),e(v8,v7)),v8),v6),v9))),v10),e(v9,v10))).
428 [hyper,90,383,284] P(e(e(x,y),e(e(y,z),e(x,z)))).
513 [hyper,90,428,226] P(e(e(x,y),e(e(e(z,x),z),y))).
611 [hyper,90,428,513] P(e(e(e(e(e(x,y),x),z),u),e(e(y,z),u))).
847 [hyper,90,611,292] P(e(e(x,y),e(y,x))).

Still focusing on *XCB*, with regard to variable richness, the best proof I know of relies on no formulas of richness strictly greater than 12, where the variable richness of a formula focuses on the greatest number of distinct variables that occurs in a deduced formula of the proof. For the curious, I do have two 22-step proofs whose respective variable richness is 16 and 19. (Yes, I suspect that Hilbert would also have been interested in the context of simplicity in variable richness.)

2.2. Seeking the Missing

Some people take keen delight in treasure hunting, even when all that is known is the object sought, but no map or clues are provided. For a charming example of this type of research, I turn to classical propositional calculus, with a focus on a treatment by Hiz in which condensed detachment is *not* a valid inference rule and is replaced with three inference rules of a similar type. The goal was to find a first-order proof (relying on three given specific inference rules) that shows his given 2-basis serves as a complete axiom system. (The first three of the following clauses encode, with hyperresolution, the inference rules, and the last two encode the 2-basis.

-P(i(x,y)) | -P(i(y,z)) | P(i(x,z)).
-P(i(x,i(y,z))) | -P(i(x,y)) | P(i(x,z)).
-P(i(n(x),y)) | -P(i(n(x),n(y))) | P(x).

P(i(n(i(x,y)),x)).
P(i(n(i(x,y)),n(y))).

The literature offered no first-order proof, and the search for such a proof had persisted for more than four decades. An appealing feature of the sought-after first-order proof is its explicit use of given inference rules, rules for drawing conclusions, and its citing of specific premisses used to deduce each step. As you most likely know, very often in mathematics no inference rule is cited and, often, the hypotheses that are used are not cited. I am confident that Hilbert would have preferred proofs with explicit inference rules (such as condensed detachment or paramodulation) being used; such proofs offer the benefit that they can be checked.

With OTTER, in early May 2004 (if I have my history correct), I sought such a proof, using as targets well-known bases. With two strategies, the game was won. To direct OTTER's reasoning, I relied on the resonance strategy, a strategy (as noted) that has the researcher include equations or formulas whose functional shape is conjectured to have much appeal. Also as noted, such items, resonators, do not have a **true** or **false** value.

Each resonator is assigned a small value to give high priority to any deduced item that matches it, where all variables are treated as indistinguishable. To restrict the reasoning, I had the program discard any deduced conclusion that relies on double negation terms, terms of the form $n(n(t))$ for some term t.

The first proof OTTER completed (in May 2004) has length 48 (applications of one of the three inference rules). Because of my constant interest in proof shortening and in keeping with Hilbert's 24th problem that focuses on simplicity, I then turned to the discovery of a "short" proof. On the next day, I succeeded, with the completion of the following, a 10-step proof.

A 10-Step Proof for Hiz

```
----- Otter 3.3d, April 2004 -----
The process was started by wos on jaguar.mcs.anl.gov,
Tue May  4 06:06:33 2004
The command was "otter".  The process ID is 12157.
-----> EMPTY CLAUSE at 38034.99 sec ----> 1712238 [hyper,9,1712132,365,349]
   $ANS(step_allLuka_1_2_3).

Length of proof is 10.  Level of proof is 5.

---------- PROOF ----------

1 [] -P(i(x,y))| -P(i(y,z))|P(i(x,z)).
2 [] -P(i(x,i(y,z)))| -P(i(x,y))|P(i(x,z)).
3 [] -P(i(n(x),y))| -P(i(n(x),n(y)))|P(x).
9 [] -P(i(i(p,q),i(i(q,r),i(p,r))))| -P(i(i(n(p),p),p))| -P(i(p,i(n(p),q)))|$ANS(step_allLuka_1_2_3).
10 [] P(i(n(i(x,y)),x)).
11 [] P(i(n(i(x,y)),n(y))).
340 [hyper,2,10,11] P(i(n(i(i(n(x),y),x)),y)).
341 [hyper,1,11,11] P(i(n(i(x,i(y,z))),n(z))).
343 [hyper,1,11,10] P(i(n(i(x,i(y,z))),y)).
349 [hyper,3,10,343] P(i(x,i(n(x),y))).
357 [hyper,1,11,343] P(i(n(i(x,i(y,i(z,u)))),z)).
365 [hyper,3,340,11] P(i(i(n(x),x),x)).
395 [hyper,1,11,341] P(i(n(i(x,i(y,i(z,u)))),n(u))).
686 [hyper,2,10,357] P(i(n(i(i(x,y),i(z,i(x,u)))),y)).
36091 [hyper,2,343,686] P(i(n(i(i(x,y),i(i(y,z),i(x,u)))),z)).
1712132 [hyper,3,36091,395] P(i(i(x,y),i(i(y,z),i(x,z)))).
```

3. The Myth of Proof Shortening and Its Pertinence to Hilbert

I come now to the promised discussion about subproofs and proof shortening. Indeed, a myth exists in this regard, a myth that is most attractive but, unfortunately, just that: a myth.

The myth asserts that one is wise to seek shorter and ever-shorter subproofs of intermediate steps or, if a conjunction is to be proved, of each of its members (or of at least one of them).

Let me demonstrate ways in which this myth is both inaccurate and dangerous.

Suppose one begins by hypothesizing a twenty-step proof whose tenth step *B* is the last step of a six-step subproof, the first five steps of which are used frequently as parents in the last half of the twenty-step (total) proof. Then imagine the discovery of a four-step proof of *B* such that its first three steps are useless for deducing any step among the last half of the total proof in focus. Finally, one can believe (as in fact sometimes is the case) that the completion of the total proof (of the theorem to be proved) that now begins with an eight-step proof of *B* results in a twenty-three step proof. Indeed, the so-to-speak replacement of five steps frequently used in the first proof by three used in no way (in the second proof) after *B* is deduced can force a person or program to include so many other steps that the second proof is far longer than the original.

My research has unearthed many such situations in which proofs of members of a conjunction are found that are shorter and still shorter while the proofs that are found of the entire conjunction get longer and still longer. Ironically, my experiments also have led to many situations in which the replacement of a shorter subproof of one member of a conjunction by a substantially longer proof of that member results in the discovery of a proof of the conjunction shorter than that relying on the shorter proof of the member in question.

So the simpler proof, in this context, is that which may rely on longer subproofs than those known to the researcher. Proof shortening is far subtler than it might at first appear to be. Indeed, one would do well to keep in mind the following aphorism: Shorter subproofs do not necessarily a shorter total proof make. And, as expected, you might like an example of this aphorism, an example that focuses on a significant result. The following story illustrates well the saying, and it focuses on an amazing (to me) proof.

The story begins with the research of great minds of the twentieth century. In particular, Lukasiewicz offered the following (shortest) single axiom for the implicational fragment of classical propositional logic.

P(i(i(i(x,y),z),i(i(z,x),i(u,x)))).

For a basis, Tarski-Bernays offered the following three formulas.

P(i(x,i(y,x))).
P(i(i(i(x,y),x),x)).
P(i(i(x,y),i(i(y,z),i(x,z)))).

To complete the background, Meredith and Prior produced a 33-step proof showing that the Lukasiewicz axiom sufficed to derive the Tarski-Bernays basis. Their proof is an abridgment of the Lukasiewicz 34-step proof. At this point—perhaps in May 2000—my colleague B. Fitelson entered the picture, suggesting that I seek a further abridgment, a proof of the theorem in focus that has length 32 or less. Of course, I was more than skeptical about the possibility, especially in view of the successes of Meredith and Prior in the context of producing "short" proofs. Nevertheless, I (as you would predict) undertook the appropriate study.

As it turned out, the key to succeeding was the use of the *cramming strategy*. Before cramming won the game, OTTER found a slightly different 33-step proof. OTTER's new 33-step proof contained a 30-step proof of the third Tarski-Bernay axiom, in contrast to the Meredith-Prior proof that contained a 31-step proof of that axiom. I focused on the new 33-step proof and asked OTTER to attempt to force, cram, those steps into proofs of both *TB*1 and *TB*2, with the hope that but two applications of condensed detachment would suffice. And that is what happened: The needed two pairs of parents were present in the 30-step proof, and OTTER had reached the goal, finding a 32-step proof, the following.

The 32-Step Proof

```
----- Otter 3.0.5b, March 1998 -----
The process was started by wos on soot.mcs.anl.gov, Wed May 24 16:24:24 2000
The command was "otter".  The process ID is 5369.

-----> EMPTY CLAUSE at   0.23 sec ----> 82 [hyper,34,53,77,79] $ANS(TARSKI_BERNAYS).

Length of proof is 32.  Level of proof is 29.

---------- PROOF ----------

33 [] -P(i(x,y)) | -P(x) | P(y).
34 [] -P(i(p,i(q,p))) | -P(i(i(i(p,q),p),p)) | -P(i(i(p,q),i(i(q,r),i(p,r)))) | $ANS(TARSKI_BERNAYS).
35 [] P(i(i(i(x,y),z),i(i(z,x),i(u,x)))).
44 [hyper,33,35,35] P(i(i(i(i(x,y),i(z,y)),i(y,u)),i(v,i(y,u)))).
45 [hyper,33,35,44] P(i(i(i(x,i(y,z)),i(i(u,y),i(v,y))),i(w,i(i(u,y),i(v,y))))).
46 [hyper,33,45,35] P(i(x,i(i(i(y,z),y),i(u,y)))).
47 [hyper,33,46,46] P(i(i(i(x,y),x),i(z,x))).
48 [hyper,33,35,47] P(i(i(i(x,y),i(y,z)),i(u,i(y,z)))).
49 [hyper,33,35,48] P(i(i(i(x,i(y,z)),i(u,y)),i(v,i(u,y)))).
50 [hyper,33,35,49] P(i(i(i(x,i(y,z)),i(u,i(z,v))),i(w,i(u,i(z,v))))).
51 [hyper,33,50,35] P(i(x,i(i(i(y,z),u),i(z,u)))).
52 [hyper,33,51,51] P(i(i(i(x,y),z),i(y,z))).
53 [hyper,33,52,52] P(i(x,i(y,x))).
55 [hyper,33,52,35] P(i(x,i(i(x,y),i(z,y)))).
56 [hyper,33,35,55] P(i(i(i(i(i(x,y),z),i(u,z)),x),i(v,x))).
57 [hyper,33,35,56] P(i(i(i(x,y),i(i(i(y,z),u),i(v,u))),i(w,i(i(i(y,z),u),i(v,u))))).
58 [hyper,33,35,57] P(i(i(i(x,i(i(i(y,z),u),i(v,u))),i(w,y)),i(v6,i(w,y)))).
59 [hyper,33,35,58] P(i(i(i(x,i(y,z)),i(u,i(i(i(z,v),w),i(v6,w)))),i(v7,i(u,i(i(i(z,v),w),i(v6,w)))))).
60 [hyper,33,59,35] P(i(x,i(i(i(y,z),i(u,v)),i(i(i(z,w),v),i(u,v))))).
61 [hyper,33,60,60] P(i(i(i(x,y),i(z,u)),i(i(i(y,v),u),i(z,u)))).
62 [hyper,33,61,35] P(i(i(i(x,y),i(z,u)),i(i(x,u),i(z,u)))).
63 [hyper,33,62,55] P(i(i(x,i(y,z)),i(i(i(x,u),z),i(y,z)))).
64 [hyper,33,63,35] P(i(i(i(i(i(x,y),z),u),i(v,x)),i(i(z,x),i(v,x)))).
65 [hyper,33,35,64] P(i(i(i(i(x,y),i(z,y)),i(i(i(y,u),x),v)),i(w,i(i(i(y,u),x),v)))).
66 [hyper,33,65,65] P(i(x,i(i(i(i(y,z),u),i(v,z)),i(i(i(z,w),v),i(y,z))))).
67 [hyper,33,66,66] P(i(i(i(i(x,y),z),i(u,y)),i(i(i(y,v),u),i(x,y)))).
68 [hyper,33,61,67] P(i(i(i(i(x,y),z),i(u,y)),i(i(i(y,v),x),i(u,y)))).
69 [hyper,33,67,68] P(i(i(i(i(x,y),z),i(i(y,u),v)),i(i(v,y),i(x,y)))).
70 [hyper,33,68,62] P(i(i(i(i(x,y),z),u),i(i(u,y),i(x,y)))).
71 [hyper,33,69,64] P(i(i(i(x,y),z),i(i(i(y,u),z),z))).
73 [hyper,33,64,71] P(i(i(x,y),i(i(i(x,z),y),y))).
75 [hyper,33,73,73] P(i(i(i(i(x,y),z),i(i(i(x,u),y),y)),i(i(i(x,u),y),y))).
76 [hyper,33,70,75] P(i(i(i(i(i(x,y),z),z),u),i(i(x,z),u))).
77 [hyper,33,75,71] P(i(i(i(x,y),x),x)).
79 [hyper,33,76,70] P(i(i(x,y),i(i(y,z),i(x,z)))).
```

This abridgment of the Meredith-Prior 33-step proof is considered by some researchers as perhaps the crowning achievement of using the cramming strategy.

As for trading longer subproofs for shorter, where the trade yields a shorter total proof, I note that the Meredith-Prior 33-step proof contains subproofs, respectively, of lengths 10, 26, and 31 of the three Tarski-Bernays axioms. As for the 32-step proof, the corresponding subproofs are of respective lengths 10, 28, and 30. In other words, a longer subproof of the second Tarski-Bernays axiom was traded for a shorter—one of length 28 for a 26-step proof—resulting in the completion of a 32-step proof in place of a 33-step

proof. I again thank Fitelson for an excellent and most enjoyable suggestion.

4. Other Types of Proof Simplification

As noted, proof shortening is only one aspect of proof simplification. And if proof shortening can be subtle—as shown with the discussion of finding shorter subproofs—proof simplification of other types can also present substantial obstacles.

Understandably, if the goal is to simplify a proof, perhaps the most obvious path is that of seeking a proof shorter than that in hand, a proof whose number of deduced steps is strictly less than that of the proof in focus. On the other hand, you might instead focus on the complexity of the proof in hand and seek a proof with less complexity, where complexity focuses on the longest (in symbols) equation or formula among the deduced steps. As you might predict, a conflict exists: a simplification with regard to length might result in a far less simple proof in the context of complexity. For example, when I was able to find a 22-step proof showing *XCB* to be a single axiom to replace (the first found) 71-step proof, I (in effect) traded (in the same order) a proof with complexity 64 for the earlier one of complexity 48. To complicate the situation, other measures of simplification exist, not the least of which is avoidance of some unwanted lemma.

In the following subsections, I focus on various aspects of proof simplification. Particular attention is given to the methodologies implemented in OTTER that have enabled me to successfully address Hilbert's twenty-fourth problem in various contexts. You find be amused to know that I worked on his 24th problem for eight years before I learned that such a problem actually existed.

4.1. Less Complex Proofs

Hilbert's concern in his 24th problem almost certainly included seeking less complex proofs. You can directly attack the problem with OTTER using the max_weight parameter. If max_weight is assigned the value k, then all conclusions that are drawn that rely on strictly more than k symbols are immediately discarded, unless the input file contains information that overrides symbol count.

In particular, you can instruct OTTER to ignore the symbol count and instead assign a user-chosen value (*weight*) to a deduced conclusion or, if that is the choice, to smaller arrays of symbols. Such assignments, each reflecting conjectured importance, are conveyed through the inclusion of patterns (whose specific variables are ignored) that correspond to fragments of equations or formulas or to entire equations or formulas. The smaller the assigned value, the greater the importance or priority as viewed by the program. Therefore, if you wish to seek a proof all of whose deduced conclusions have a complexity that does not exceed a chosen k, you simply assign the value k to max_weight and avoid the inclusion of weight templates that could interfere with your wishes. On the other hand, if you wish to permit the use in a proof of one or more equations or formulas whose complexity does exceed k, OTTER offers what is needed. Such can occur, for example, when you have a favorite lemma to be included in proofs or "know" that some equation or formula, though complex, is required for the completion of a sought-after proof.

My experiments with seeking proofs of smaller complexity than those found in the literature have been repeatedly successful. One of the more satisfying studies led to the discovery of a proof of complexity 47, in contrast to the original proof of complexity 103. In addition, although not the primary goal of the experiment, by limiting the complexity of retained conclusions, I have occasionally discovered a proof shorter than offered by the literature. On the other hand, sometimes a tradeoff occurs: The program finds a proof of less complexity, but one of greater length; the reverse trade was presented in the preceding section.

And of course, sometimes a "messy" step may be required for the completion of a proof. Indeed, I am almost certain that the simplest proof of the sufficiency of the formula *XCB* as a single axiom for equivalential calculus relies on formulas of complexity 47 (measured in symbol count, ignoring the predicate symbol). (I have in fact proofs of this nature, one of which requires only a single such formula.)

For a more impressive example in the context of large formulas, one of my studies focused on what some call a Rezus-style formula, sometimes credited to Tarski. To deduce that formula, a new strategy was formulated, namely, the *subformula strategy*. You see, the formula (ignoring predicate symbol) has length

93. It was clear that unless a new strategy was introduced, OTTER would almost never consider formulas of even two-thirds that length.

4.2. Variable Richness

Another aspect of proof simplification that might indeed have interested Hilbert is variable richness. Logicians prefer axiom systems in which fewer letters, or variables, are present than more. In particular, they show more interest in the axiom system *A* over the axiom system *B*, ceteris paribus, if the maximum number of distinct variables in the members of *A* (considered separately) is strictly less than that for *B*. You might say, in this case, that the system *A* is simpler than is *B*. A natural extension of this notion asserts that the proof *P* is simpler than the proof *Q* if and only if the maximum number of distinct variables for the deduced steps of *P* (considered one at a time) is strictly less than is the case for *Q*.

OTTER offers the means for directly attacking this aspect of proof simplification. Indeed, one can assign to the parameter max_distinct_vars a value *k*, which has the program discard any newly deduced conclusion if the number of distinct variables on which it relies strictly exceeds *k*. I have successfully conducted experiments in various areas of logic with the goal of reducing variable richness. Compared with what might be burdensome for even a master unaided by automation, addressing this aspect of simplification with an automated reasoning program is most pleasant.

In some cases, one can satisfy a Hilbert goal regarding the ideal or limiting situation; indeed, I have showed that no proof of the theorem under study exists with a variable richness of *j* or less. In other studies, I have found proofs with a variable richness less than that offered by the literature. For example, Meredith's proof, of length 41, for his single axiom (the following) for classical propositional calculus has variable richness 7.

P(i(i(i(i(i(x,y),i(n(z),n(u))),z),v),i(i(v,x),i(u,x)))).

D. Kapur (perhaps on May 27, 1994) found a 69-step proof with OTTER with richness 6. I later (perhaps on September 9, 2000) based on his proof as resonators (if memory serves) found a 49-step proof with variable richness 6. Still later (perhaps on September 11, 2001) I found a 5-variable proof (and, as you might expect, the proof is far longer than is Meredith's, length 68 versus 41). A 4-variable proof does not exist, shown with OTTER by assigning the value 4 to max_distinct_vars; indeed, with that value no conclusions (other than the axiom) can be drawn with the use of condensed detachment. Meredith's proof, Kapur's proof, and my 5-variable proof each conclude with the derivation of the well-known 3-axiom system of Lukasiewicz for this area of logic, the following.

P(i(i(x,y),i(i(y,z),i(x,z)))).
P(i(i(n(x),x),x)).
P(i(x,i(n(x),y))).

You thus have in hand a typical tradeoff, in this case, more simplicity in the context of variable richness traded for far less simplicity in the context of proof length. After all, the restriction to five variables for deduced conclusions to be used places a severe constraint, causing the program to use many more formulas than Meredith used.

You might then immediately ask about Meredith, his interest in short proofs, and whether his 41-step proof is the shortest known. Yes, Meredith was indeed interested in finding short proofs, but his 41-step proof is not, any longer, the shortest known. Specifically, after searching on and off for eight years, I found a 38-step proof. As pointed out by my colleague Fitelson, that proof offers piquancy, namely, in the order of the three Lukasiewicz axioms proved: 3, 1, and 2. The piquancy rests with the fact that Lukasiewicz's first axiom is ordinarily far more difficult to prove than the other two axioms. Yet in the 38-step proof, Lukasiewicz axiom 3 is proved first, then Lukasiewicz axiom 1, and finally Lukasiewicz axiom 2.

A 38-Step Proof for the Meredith Single Axiom

----- Otter 3.1-b0, May 2000 -----

The process was started by wos on lemma.mcs.anl.gov,
Sun Sep 10 10:02:51 2000
The command was "otter". The process ID is 10751.
-----> EMPTY CLAUSE at 2082.87 sec ----> 229062 [hyper,7,228420,228800,96]
$ANS(step_allLuka_1_2_3).

Length of proof is 38. Level of proof is 25.

---------- PROOF ----------

1 [] -P(i(x,y))| -P(x)|P(y).
7 [] -P(i(i(p,q),i(i(q,r),i(p,r))))| -P(i(i(n(p),p),p))| -P(i(p,i(n(p),q)))|$ANS(step_allLuka_1_2_3).
8 [] P(i(i(i(i(i(x,y),i(n(z),n(u))),z),v),i(i(v,x),i(u,x)))).
26 [hyper,1,8,8] P(i(i(i(i(x,y),i(z,y)),i(y,u)),i(v,i(y,u)))).
28 [hyper,1,8,26] P(i(i(i(x,i(n(y),z)),u),i(y,u))).
30 [hyper,1,8,28] P(i(i(i(x,x),y),i(z,y))).
32 [hyper,1,30,30] P(i(x,i(y,i(z,z)))).
48 [hyper,1,8,32] P(i(i(i(x,i(y,y)),z),i(u,z))).
54 [hyper,1,8,48] P(i(i(i(x,y),z),i(y,z))).
82 [hyper,1,8,54] P(i(i(i(i(n(x),n(y)),x),z),i(y,z))).
85 [hyper,1,54,8] P(i(x,i(i(x,y),i(z,y)))).
95 [hyper,1,8,82] P(i(i(i(x,y),n(i(n(y),n(z)))),i(z,n(i(n(y),n(z))))).
96 [hyper,1,82,54] P(i(x,i(n(x),y))).
103 [hyper,1,54,85] P(i(x,i(i(i(y,x),z),i(u,z)))).
123 [hyper,1,85,95] P(i(i(i(i(i(x,y),n(i(n(y),n(z)))),i(z,n(i(n(y),n(z))))),u),i(v,u))).
134 [hyper,1,54,96] P(i(x,i(n(i(y,x)),z))).
140 [hyper,1,28,96] P(i(x,i(n(i(y,i(n(x),z))),u))).
169 [hyper,1,8,103] P(i(i(i(i(i(x,i(i(i(y,z),i(n(u),n(v))),u)),w),i(v6,w)),y),i(v,y))).
189 [hyper,1,8,123] P(i(i(i(x,y),i(z,u)),i(i(n(u),n(n(y))),i(z,u)))).
223 [hyper,1,54,134] P(i(x,i(n(i(y,i(z,x))),u))).
320 [hyper,1,8,169] P(i(i(i(x,y),i(z,i(i(i(y,u),i(n(v),n(x))),v))),i(w,i(z,i(i(i(y,u),i(n(v),n(x))),v))))).
337 [hyper,1,189,123] P(i(i(n(x),n(n(x))),i(y,x))).
343 [hyper,1,189,8] P(i(i(n(i(x,y)),n(n(z))),i(i(z,y),i(x,y)))).
428 [hyper,1,320,85] P(i(x,i(i(i(y,z),u),i(i(i(z,v),i(n(u),n(y))),u)))).
437 [hyper,1,54,337] P(i(n(n(x)),i(y,x))).
451 [hyper,1,428,428] P(i(i(i(x,y),z),i(i(i(y,u),i(n(z),n(x))),z))).
503 [hyper,1,451,8] P(i(i(i(x,y),i(n(i(i(x,z),i(u,z))),n(i(i(i(z,v),i(n(w),n(u))),w)))),i(i(x,z),i(u,z)))).
511 [hyper,1,503,223] P(i(i(x,i(x,y)),i(z,i(x,y)))).
512 [hyper,1,503,140] P(i(i(x,y),i(n(i(x,z)),y))).
528 [hyper,1,511,437] P(i(x,i(n(n(y)),y))).
606 [hyper,1,85,528] P(i(i(i(x,i(n(n(y)),y)),z),i(u,z))).
644 [hyper,1,511,606] P(i(x,i(i(i(y,i(n(n(z)),z)),u),u))).
227886 [hyper,1,644,644] P(i(i(i(i(x,i(n(n(y)),y)),z),z),z)).
227915 [hyper,1,512,227886] P(i(n(i(i(i(x,i(n(n(y)),y)),z),u)),z)).
227916 [hyper,1,451,227886] P(i(i(i(i(x,y),i(n(x),n(i(z,i(n(n(u)),u))))),x)).
227962 [hyper,1,343,227915] P(i(i(i(x,y),i(i(i(z,i(n(n(u)),u)),n(n(x))),y))).
228080 [hyper,1,227962,227962] P(i(i(i(i(x,i(n(n(y)),y)),n(n(i(z,u)))),i(i(i(v,i(n(n(w)),w)),n(n(z))),u))).
228289 [hyper,1,8,228080] P(i(i(i(i(i(x,i(n(n(y)),y)),n(n(z))),u),v),i(i(z,u),v))).
228420 [hyper,1,228289,8] P(i(i(x,y),i(i(y,z),i(x,z)))).
228423 [hyper,1,228420,228420] P(i(i(i(i(x,y),i(z,y)),u),i(i(z,x),u))).
228800 [hyper,1,228423,227916] P(i(i(n(x),x),x)).

4.3. Term Avoidance

An aspect of proof simplification that might easily escape notice concerns the type of term present in a proof. For example, if the literature offers a proof of a deep theorem in group theory in which, among the deduced steps, you will find expressions of the form "inverse of inverse of *t*" for some term *t*, there might exist a simpler proof, simpler because of avoiding such terms. Similarly, in logic, the avoidance of double-negation terms, those of the form $n(n(t))$ for some term *t*, might result in the discovery of a simpler proof with the focus on the type of term present. A program such as OTTER can be instructed to search for the type of proof in focus, for example, a double-negation-free proof. With OTTER, you can avoid such terms either with the nonstandard use of demodulators (rewrite rules), by rewriting unwanted expressions to "junk", or with the use of weighting, by assigning to unwanted terms a degree of complexity strictly greater than the value assigned to max_weight.

Among my successes with term avoidance is the discovery of a double-negation-free proof for a theorem of Meredith showing that one of the five axioms offered by Lukasiewicz for his infinite-valued sentential calculus is in fact dependent. When I presented this proof to colleagues, they evidenced great surprise in that the literature appears to suggest that double-negation terms are needed. Further, as a fine example of how the research of one individual can affect that of another, my continued success in the discovery of one double-negation-free proof after another indirectly led to a question by Ulrich. Specifically, does there exist an axiom system for propositional calculus such that, when the theorem to be proved is free of double negation, you can always find a double-negation-free proof? That question was answered in the affirmative by M. Beeson and R. Veroff when it was shown that the well-known three-axiom system of Lukasiewicz has the desired property. The Beeson-led team also proved similar theorems for other areas of logic.

Sometimes you can find a proof that is simpler than the literature offers and simpler in more than one way. Indeed, I eventually found a proof showing one of the five Lukasiewicz axioms (his fifth) for his infinite-valued logic is dependent, a proof of length 30 (applications of condensed detachment, avoiding the use of double-negation terms, and avoiding relying on (as noted earlier) three lemmas that Rose and Rosser might have thought crucial.

Here are the first four Lukasiewicz axioms, then the fifth to be proved, then the three lemmas (2.22, 3.5, 3.51) to be avoided and, finally, the 30-step proof that avoids the three lemmas and avoids the use of double negation.

```
P(i(x,i(y,x))).
P(i(i(x,y),i(i(y,z),i(x,z)))).
P(i(i(i(x,y),y),i(i(y,x),x))).
P(i(i(n(x),n(y)),i(y,x))).

P(i(i(i(x,y),i(y,x)),i(y,x))).

P(i(i(i(x,y),i(z,y)),i(i(y,x),i(z,x)))).
P(i(i(x,y),i(n(y),n(x)))).
P(i(i(i(x,y),i(x,z)),i(i(y,x),i(y,z)))).
```

A 30-Step Proof of a Theorem of Meredith

```
----- Otter 3.0.5b, March 1998 -----
The process was started by wos on ember.mcs.anl.gov, Sat May 19 09:58:37 2001
The command was "otter".  The process ID is 7716.

----> UNIT CONFLICT at   1.90 sec ----> 2456 [binary,2455.1,6.1] $ANS(MV_5).

Length of proof is 30.  Level of proof is 21.
```

---------- PROOF ----------

1 [] -P(i(x,y))| -P(x)|P(y).
2 [] P(i(x,i(y,x))).
3 [] P(i(i(x,y),i(i(y,z),i(x,z)))).
4 [] P(i(i(i(x,y),y),i(i(y,x),x))).
5 [] P(i(i(n(x),n(y)),i(y,x))).
6 [] -P(i(i(i(a,b),i(b,a)),i(b,a)))|$ANS(MV_5).
21 [hyper,1,3,3] P(i(i(i(i(x,y),i(z,y)),u),i(i(z,x),u))).
23 [hyper,1,3,2] P(i(i(i(x,y),z),i(y,z))).
26 [hyper,1,3,5] P(i(i(i(x,y),z),i(i(n(y),n(x)),z))).
28 [hyper,1,21,21] P(i(i(x,i(y,z)),i(i(u,y),i(x,i(u,z))))).
37 [hyper,1,23,4] P(i(x,i(i(x,y),y))).
42 [hyper,1,3,26] P(i(i(i(i(n(x),n(y)),z),u),i(i(i(y,x),z),u))).
44 [hyper,1,26,23] P(i(i(n(x),n(i(y,z))),i(z,x))).
70 [hyper,1,28,37] P(i(i(x,i(y,z)),i(y,i(x,z)))).
74 [hyper,1,3,37] P(i(i(i(i(x,y),y),z),i(x,z))).
126 [hyper,1,3,70] P(i(i(i(x,i(y,z)),u),i(i(y,i(x,z)),u))).
145 [hyper,1,126,28] P(i(i(x,i(y,z)),i(i(u,x),i(y,i(u,z))))).
162 [hyper,1,145,4] P(i(i(x,i(i(y,z),z)),i(i(z,y),i(x,y)))).
163 [hyper,1,145,3] P(i(i(x,i(y,z)),i(i(z,u),i(x,i(y,u))))).
202 [hyper,1,163,162] P(i(i(i(x,y),z),i(i(x,i(i(y,u),u)),i(i(u,y),z)))).
293 [hyper,1,202,5] P(i(i(n(x),i(i(n(y),z),z)),i(i(z,n(y)),i(y,x)))).
781 [hyper,1,42,23] P(i(i(i(x,y),z),i(n(x),z))).
806 [hyper,1,3,781] P(i(i(i(n(x),y),z),i(i(i(x,u),y),z))).
952 [hyper,1,806,293] P(i(i(i(x,y),i(i(n(z),u),u)),i(i(u,n(z)),i(z,x)))).
1063 [hyper,1,952,5] P(i(i(x,n(y)),i(y,n(x)))).
1102 [hyper,1,145,1063] P(i(i(x,i(y,n(z))),i(z,i(x,n(y))))).
1125 [hyper,1,806,1102] P(i(i(i(x,y),i(z,n(u))),i(u,i(n(x),n(z))))).
1283 [hyper,1,21,1125] P(i(i(x,y),i(z,i(n(y),n(x))))).
1564 [hyper,1,162,1283] P(i(i(i(n(x),n(y)),z),i(i(y,x),z))).
1766 [hyper,1,74,1564] P(i(n(x),i(i(y,x),n(y)))).
1791 [hyper,1,1102,1766] P(i(x,i(n(y),n(i(x,y))))).
2016 [hyper,1,145,1791] P(i(i(x,y),i(n(z),i(x,n(i(y,z)))))).
2243 [hyper,1,2016,44] P(i(n(x),i(i(n(y),n(i(z,u))),n(i(i(u,y),x))))).
2303 [hyper,1,293,2243] P(i(i(n(i(i(x,y),x)),n(y)),i(y,x))).
2366 [hyper,1,1564,2303] P(i(i(x,i(i(y,x),y)),i(x,y))).
2455 [hyper,1,126,2366] P(i(i(i(x,y),i(y,x)),i(y,x))).

The focus on such avoidance brings this notebook to the next section.

4.4. Lemma Avoidance

Still in the spirit of avoidance, but with the emphasis on lemma avoidance, as noted, the cited 30-step dependence proof in infinite-valued logic also avoids reliance on three lemmas that appeared to be indispensable. Finally, as an example of having won in many areas simultaneously—a most unlikely occurrence made possible because of the use of OTTER—the cited proof is the shortest of which I know, of length 30 (applications of condensed detachment). For an intriguing question, does there exist a proof of length 29 or less that shows that the fifth axiom of Lukasiewicz (for infinite-vaued sentential calculus) is dependent on the remaining four. As you might expect, I am very interested in such a proof, whether or not it avoids various lemmas or avoids the use of double negation.

5. Hypothesis Testing and a First Step toward Hilbert's Desired Theory

An automated reasoning program can serve you well in hypothesis testing. Indeed, my study of double-negation-free proofs provides a fine example. As noted, a double-negation-free proof is a proof none of whose steps contains a term of the form $n(n(t))$ for some term t, where n denotes negation. In this sense, clearly such a proof is "simpler" than proofs in which double-negation terms are present. Years after I had found the first such proof—the context was Lukasiewicz's infinite-valued sentential calculus—my colleagues and I sought similar proofs for other theorems, focusing initially on classical propositional calculus. I hypothesized that one could always find a double-negation-free proof when both the axiom system and the theorem were free of double-negation terms. To test the hypothesis, I conducted one experiment after another, focusing on one theorem after another, to see whether OTTER could always find a proof with the desired properties.

Meeting but two failures, I then posed the question, What conditions guarantee that a double-negation-free proof must exist? Shortly thereafter, Ulrich posed the related question about the existence of an axiom system for classical propositional calculus with the desired properties. Beeson wrote a charming program that found double-negation proofs for my two failures, for the respective theorems on focus. The two questions were answered by my colleagues Beeson and Veroff. In short, the verification (with OTTER) of the hypothesis was followed by a delightful metatheorem.

I view this as a step in the right direction—the direction of answering Hilbert's question of developing a theory about proof simplicity.

6. Open Questions and Explicit Challenges

At this point, I offer several open questions and challenges, many of which are taken from Chapter 7 of *Automated Reasoning and the Discovery of Missing and Elegant Proofs*; for your convenience, I shall frequently employ notation taken from that boo. I also use the numbering scheme from that book (e.g., OQ20.PC means Open Question number 20 in the Propositional Calculus subsection); my intention is to be consistent in referring to specific challenges and open questions, rather than giving them different numbers in different manuscripts. (If you wish to find more challenges and open questions, Chapter 7 is one source; my various notebooks found on my website is another good source.) Some of the challenges and questions, as can easily be seen, are of the type that the Hilbert 24th problem focuses on. The likelihood of success in considering one of these questions and challenges is increased when an automated reasoning program is part of the team—even more so when the program offers a number of inference rules and, more important, numerous strategies, some to restrict the reasoning, some to direct it, and others. Of course, the open questions and chalenges you now are offered are tough ones. However, your attempt to dispatch any of them might bring you intrigue and excitement. First, before offering the possible treasure, a very small taste of rules for drawing conclusions and means for controlling such rules might serve you well.

Regarding inference rules, when equality is present, one can (as many of you know) rely on paramodulation, a rule that generalizes equality substitution; its use is seen in a proof from lattice theory to be given shortly. This rule, from what I can tell, is not used by mathematicians or logicians probably because of its unexpected nature and its complexity. Specifically, in contrast to what ordinarily occurs, with this inference rule, one is permitted to replace variables in a nontrivial way in both the *from* and the *into* hypotheses (parents). Indeed, this inference rule was formulated to take advantage of the nature of computing, not to emulate what a person does. Paramodulation proves to be a powerful inference rule for areas such as lattice theory, group theory, and abstract algebra in general.

Quite different, and familiar to many, is the inference rule hyperresolution, useful, for example, when condensed detachment is in focus. The rule considers two or more hypotheses, requiring that the conclusion be free of logical **not**. When the intent is to draw nonempty conclusions free of logical **or**, then the rule of choice is UR-resolution.

For restricting the program's reasoning, the set of support strategy is perhaps the most powerful to use. With that strategy, you partition the input statements in such a manner that, usually, the so-called special hypothesis, and perhaps the denial of the conclusion, are placed on a list used by the program to initiate lines of reasoning. An example of a special hypothesis is provided by the equation $xxx = x$, when the area

of study is rings in which the cube of x is x. To direct the reasoning, both the resonance strategy and Veroff's hints strategy have proved extremely useful. They can be used at the same time, or each can be used on its own. Essentially, each of the two strategies asks the researcher to include equations or formulas that are conjectured to merit particular emphasis. In effect, to each deduced inclusion that matches an included resonator or hint, you assign a high priority for being used to initiate a line of reasoning.

If you browse in various of my notebooks, you will discover other items that may enable you to reach your goal, regardless of its nature.

6.1. Lattice Theory

The first five questions and challenges come from the area (or variety) of mathematics known as lattice theory. This variety can be axiomatized in terms of join and meet, where "v" denotes join (or union) and "^" meet (or intersection). McCune's excellent study of this area of mathematics focused mainly on the following 4-basis of R. McKenzie.

y v (x ^ (y ^ z)) = y.
y ^ (x v (y v z)) = y.
((x ^ y) v (y ^ z)) v y = y.
((x v y) ^ (y v z)) ^ y = y.

Among the results, McCune's research culminated in the discovery of the following two short single axioms, each of length 29.

(((y v x) ^ x) v (((z ^ (x v x)) v (u ^ x)) ^ v)) ^ (w v ((v6 v x) ^ (x v v7))) = x.
(((y v x) ^ x) v (((z ^ (x v x)) v (u ^ x)) ^ v)) ^ (((w v x) ^ (v6 v x)) v v7) = x.

OQ09.LT: Given that the shortest found by McCune has length 29, does there exist a shorter single axiom for lattice theory in terms of meet and join?

CH07.LT: Are the given 29-letter single axioms for lattice theory the only ones of that length?

OQ10.LT: In terms of meet and join, how long is the shortest single axiom for lattice theory?

CH08.LTa: Does there exist a proof of length strictly less than 42 (applications of paramodulation) that deduces the given 4-basis from the first of the two 29-letter single axioms for lattice theory such that the proof relies solely on forward reasoning and does not rely on demodulation (to automatically simplify and canonicalize)? If you consult the cited book of mine, you will find the number 50 in place of 42, and you then might wonder why. Well, on March 8, 2002, pursuant to a request from McCune, I did find a 50-step proof, which was the shortest at the time and which pleased him much. I used cramming and many other strategies and approaches but could get no further. Then, as I was writing my notebook on algebra in 2009, I decided to try again for a shorter proof focusing on McCune's first of his two 29-symbol single axioms for lattice theory. On September 7 2009, with cramming and a small assigned value to max_weight and other changes that I am unable to fully document, after various experiments, OTTER offered me a 42-step proof, the following.

A 42-Step Proof for McCune's First Single Axiom for Lattices

----- Otter 3.3g-work, Jan 2005 -----
The process was started by wos on crush.mcs.anl.gov,
Wed Jul 8 19:26:14 2009
The command was "otter". The process ID is 6641.

-----> EMPTY CLAUSE at 4.39 sec ----> 3939 [hyper,3647,2,2395,1144,3103] $ANS(step_all).

Length of proof is 42. Level of proof is 28.

---------- PROOF ----------

2 [] b v (a^ (b^c))!=b|b^ (a v (b v c))!=b| ((a^b) v (b^c)) v b!=b| ((a v b)^ (b v c))^b!=b|$ANS(step_all).
4 [] (((y v x)^x) v (((z^ (x v x)) v (u^x))^v))^ (w v ((v6 v x)^ (x v v7)))=x.
13 [para_into,4.1.1.1.2,4.1.1] (((x v y)^y) v (y v y))^ (z v ((u v y)^ (y v v)))=y.
14 [para_into,13.1.1.2.2,13.1.1] (((x v (y v y))^ (y v y)) v ((y v y) v (y v y)))^ (z v y)=y v y.
23 [para_from,14.1.1,4.1.1.1.2.1.1] (((x v y)^y) v (((y v y) v (z^y))^u))^ (v v ((w v y)^ (y v v6)))=y.
38 [para_from,23.1.1,13.1.1.2.2] (((x v (((y v y) v (z^y))^u))^ (((y v y) v (z^y))^u)) v
((((y v y) v (z^y))^u) v (((y v y) v (z^y))^u)))^ (v v y)= ((y v y) v (z^y))^u.
49 [para_from,38.1.1,4.1.1.1.2.1.1] (((x v y)^y) v (((((y v y) v (z^y))^u) v (v^y))^w))^
(v6 v ((v7 v y)^ (y v v8)))=y.
56 [para_into,49.1.1.1.2,49.1.1] (((x v y)^y) v (z^y))^ (u v ((v v y)^ (y v w)))=y.
68 [para_into,56.1.1.2.2,56.1.1] (((x v (y^z))^ (y^z)) v (u^ (y^z)))^ (v v z)=y^z.
91 [para_from,68.1.1,4.1.1.1.2.1.1] (((x v y)^y) v (((z^y) v (u^y))^v))^ (w v ((v6 v y)^ (y v v7)))=y.
114 [para_into,91.1.1.1.2,91.1.1] (((x v y)^y) v y)^ (z v ((u v y)^ (y v v)))=y.
136 [para_from,114.1.1,91.1.1.2.2] (((x v y)^y) v (((z^y) v (u^y))^v))^ (w v y)=y.
137 [para_from,114.1.1,56.1.1.2.2] (((x v y)^y) v (z^y))^ (u v y)=y.
140 [para_from,114.1.1,13.1.1.2.2] (((x v y)^y) v (y v y))^ (z v y)=y.
243 [para_from,140.1.1,56.1.1.1.1] (x v (y^ (x v x)))^ (z v ((u v (x v x))^ ((x v x) v v)))=x v x.
251 [para_into,243.1.1.1.2,140.1.1] (x v x)^ (y v ((z v (x v x))^ ((x v x) v u)))=x v x.
327 [para_from,251.1.1,136.1.1.1.2] (((x v y)^y) v ((z^y) v (z^y)))^ (u v y)=y.
418 [para_from,327.1.1,251.1.1.2.2] ((x^y) v (x^y))^ (z v y)= (x^y) v (x^y).
444 [para_into,418.1.1,137.1.1,flip.1] ((x v y)^y) v ((x v y)^y)=y.
575 [para_from,444.1.1,91.1.1.1.2.1] (((x v y)^y) v (y^z))^ (u v ((v v y)^ (y v w)))=y.
579 [para_from,444.1.1,56.1.1.1] x^ (y v ((z v x)^ (x v u)))=x.
716 [para_into,579.1.1.2.2,68.1.2] x^ (y v ((((z v ((u v x)^ (x v v)))^ ((u v x)^ (x v v))) v
(w^ ((u v x)^ (x v v))))^ (v6 v (x v v))))=x.
738 [para_from,579.1.1,136.1.1.1.2] (((x v y)^y) v ((z^y) v (u^y)))^ (v v y)=y.
938 [para_from,738.1.1,579.1.1.2.2] ((x^y) v (z^y))^ (u v y)= (x^y) v (z^y).
1111 [para_into,938.1.1,137.1.1,flip.1] ((x v y)^y) v (z^y)=y.
1144 [para_from,1111.1.1,716.1.1.2] x^ (y v (x v z))=x.
1155 [para_from,1111.1.1,575.1.1.2] (((x v y)^y) v (y^z))^ (y v u)=y.
1207 [para_into,1155.1.1,579.1.1] ((x v y)^y) v (y^z)=y.
1282 [para_into,1207.1.1.2,579.1.1] ((x v y)^y) v y=y.
1292 [para_from,1207.1.1,68.1.1.2] (((x v (y^ (z^u)))^ (y^ (z^u))) v (v^ (y^ (z^u))))^z=y^ (z^u).
1314 [para_from,1282.1.1,579.1.1.2] x^ ((y v x)^ (x v z))=x.
1343 [para_from,1282.1.1,1111.1.1.1.1] (x^x) v (y^x)=x.
1454 [para_into,1314.1.1.2,575.1.1] (x^y)^x=x^y.
1460 [para_into,1314.1.1.2,114.1.1] x^x=x.
1619 [para_from,1454.1.1,738.1.1.1.2.2] (((x v y)^y) v ((z^y) v (y^u)))^ (v v y)=y.
1803 [para_from,1460.1.1,1343.1.1.1] x v (y^x)=x.
1823 [para_from,1460.1.1,738.1.1.1.2.2] (((x v y)^y) v ((z^y) v y))^ (u v y)=y.
2045 [para_from,1619.1.1,1314.1.1.2] ((x^y) v (y^z))^y= (x^y) v (y^z).
2393 [para_into,1803.1.1.2,1314.1.1] ((x v y)^ (y v z)) v y= (x v y)^ (y v z).
2395 [para_into,1803.1.1.2,1292.1.1] x v (y^ (x^z))=x.
2415 [para_into,1803.1.1,1207.1.1,flip.1] (x v y)^y=y.
3103 [para_into,2415.1.1.1,2393.1.1] ((x v y)^ (y v z))^y=y.
3123 [para_into,2415.1.1,1823.1.1,flip.1] (x^y) v y=y.
3647 [para_into,3123.1.1.1,2045.1.1] ((x^y) v (y^z)) v y=y.

So, yes, I worked on one of the challenges I had posed years earlier.

Then, in keeping with Hilbert's interest in simpler proofs, I sought a proof whose variable richness is 8; the cited 42-step proof has variable richness 9. On September 27 of the same year, I found a 72-step proof. When I sought a proof with richness 7, I found a 74-step proof,, but five minutes later. So I am now presenting an additional challenge (in the context of length) focusing on McCune's first 29-letter single axiom for lattice theory. I believe that no proof of richness 6 exists, but I am not certain.

CH09.LTa: Does there exist a short proof (say, of length 54 or less) with the second 29-letter single axiom as sole hypothesis that completes with the given 4-basis for lattice theory, where the proof is required to rely solely on forward reasoning and to avoid the use of demodulation? As in the preceding challenge, the challenge is a bit different from that offered in my book. You see, for years, I knew of no short proof. However, about four years before writing the algebra notebook, I tried to meet the challenge presented in the book, namely, finding a proof of length 60 or less. Eventually, OTTER produced a 55-step proof, the following.

A 55-Step Proof for McCune's Second Single Axiom for Lattice Theory

```
----- Otter 3.3g-work, Jan 2005 -----
The process was started by wos on jaguar.mcs.anl.gov,
Sat Feb 26 01:23:33 2005
The command was "otter".  The process ID is 30103.
-----> EMPTY CLAUSE at   0.85 sec ----> 3346 [hyper,3025,2,2716,2804,1483] $ANS(step_all).

Length of proof is 55.  Level of proof is 32.

---------- PROOF ----------

2 [] b v (a^ (b^c))!=b|b^ (a v (b v c))!=b| ((a^b) v (b^c)) v b!=b| ((a v b)^ (b v c))^b!=b|$ANS(step_all).
4 [] (((y v x)^x) v (((z^ (x v x)) v (u^x))^v))^ (((w v x)^ (v6 v x)) v v7)=x.
65 [para_into,4.1.1.1.2,4.1.1] (((x v y)^y) v (y v y))^ (((z v y)^ (u v y)) v v)=y.
67 [para_from,65.1.1,4.1.1.2.1] (((x v (y v y))^ (y v y)) v (((z^ ((y v y) v (y v y)))
  v (u^ (y v y)))^v))^ (y v w)=y v y.
68 [para_into,67.1.1.1.2.1.2,67.1.1] (((x v (y v y))^ (y v y)) v (((z^ ((y v y) v (y v y)))
  v (y v y))^u))^ (y v v)=y v y.
69 [para_from,67.1.1,4.1.1.1.2.1.1] (((x v y)^y) v (((y v y) v (z^y))^u))^ (((v v y)^ (w v y)) v v6)=y.
71 [para_from,69.1.1,65.1.1.2.1] (((x v (((y v y) v (z^y))^u))^ (((y v y) v (z^y))^u)) v ((((y v y)
  v (z^y))^u) v (((y v y) v (z^y))^u)))^ (y v v)= ((y v y) v (z^y))^u.
72 [para_from,71.1.1,4.1.1.1.2.1.1] (((x v y)^y) v (((((y v y) v (z^y))^u) v (v^y))^w))
  ^ (((v6 v y)^ (v7 v y)) v v8)=y.
73 [para_into,72.1.1.1.2,72.1.1] (((x v y)^y) v (z^y))^ (((u v y)^ (v v y)) v w)=y.
76 [para_into,73.1.1.2.1,73.1.1] (((x v (y^z))^ (y^z)) v (u^ (y^z)))^ (z v v)=y^z.
82 [para_from,76.1.1,4.1.1.1.2.1.1] (((x v y)^y) v (((z^y) v (u^y))^v))^ (((w v y)^ (v6 v y)) v v7)=y.
83 [para_into,82.1.1.1.2,82.1.1] (((x v y)^y) v y)^ (((z v y)^ (u v y)) v v)=y.
89 [para_from,83.1.1,82.1.1.2.1] (((x v y)^y) v (((z^y) v (u^y))^v))^ (y v w)=y.
90 [para_from,83.1.1,73.1.1.2.1] (((x v y)^y) v (z^y))^ (y v u)=y.
93 [para_from,83.1.1,65.1.1.2.1] (((x v y)^y) v (y v y))^ (y v z)=y.
105 [para_from,89.1.1,73.1.1.1.2] (((x v (y v z))^ (y v z)) v y)^ (((u v (y v z))^ (v v (y v z))) v w)=y v z.
124 [para_from,93.1.1,68.1.1.1.1] (x v (((y^ ((x v x) v (x v x))) v (x v x))^z))^ (x v u)=x v x.
132 [para_into,105.1.1.1.1,93.1.1] (x v x)^ (((y v (x v x))^ (z v (x v x))) v u)=x v x.
146 [para_into,124.1.1.1.2.1.1,76.1.1] (x v (((y^ (x v x)) v (x v x))^z))^ (x v u)=x v x.
148 [para_from,124.1.1,89.1.1.1.2] (((x v y)^y) v ((z^y) v (z^y)))^ (y v u)=y.
171 [para_into,146.1.1.1.2.1.1,76.1.1] (x v (((y^x) v (x v x))^z))^ (x v u)=x v x.
181 [para_from,148.1.1,132.1.1.2.1] ((x^y) v (x^y))^ (y v z)= (x^y) v (x^y).
201 [para_into,181.1.1,90.1.1,flip.1] ((x v y)^y) v ((x v y)^y)=y.
```

245 [para_from,201.1.1,73.1.1.1] x^ (((y v x)^ (z v x)) v u)=x.
265 [para_into,245.1.1.2.1,105.1.1] x^ ((x v y) v z)=x.
276 [para_from,245.1.1,89.1.1.1.2] (((x v y)^y) v ((z^y) v (u^y)))^ (y v v)=y.
316 [para_from,276.1.1,245.1.1.2.1] ((x^y) v (z^y))^ (y v u)= (x^y) v (z^y).
328 [para_into,316.1.1,90.1.1,flip.1] ((x v y)^y) v (z^y)=y.
364 [para_into,328.1.1.2,276.1.1] ((x v (y v z))^ (y v z)) v y=y v z.
438 [para_from,364.1.1,328.1.1.1.1] ((x v y)^x) v (z^x)=x.
500 [para_from,438.1.1,245.1.1.2] x^ (y v x)=x.
548 [para_into,500.1.1,83.1.1,flip.1] ((x v y)^y) v y=y.
572 [para_into,548.1.1.1.1,548.1.1] (x^x) v x=x.
590 [para_from,548.1.1,500.1.1.2] x^x=x.
654 [para_from,590.1.1,572.1.1.1] x v x=x.
700 [para_into,654.1.1,328.1.1,flip.1] (x v y)^y=y.
716 [para_from,654.1.1,245.1.1.2] x^ ((y v x)^ (z v x))=x.
722 [para_from,654.1.1,171.1.1.1.2.1.2] (x v (((y^x) v x)^z))^ (x v u)=x v x.
731 [para_from,654.1.1,316.1.1.2] ((x^y) v (z^y))^y= (x^y) v (z^y).
788 [para_into,700.1.1.1,438.1.1] x^ (y^x)=y^x.
940 [para_into,722.1.2,654.1.1] (x v (((y^x) v x)^z))^ (x v u)=x.
1075 [para_into,788.1.1.2,716.1.1] ((x v y)^ (z v y))^y=y^ ((x v y)^ (z v y)).
1320 [para_into,940.1.1,590.1.1] x v (((y^x) v x)^z)=x.
1395 [para_into,1075.1.2,716.1.1] ((x v y)^ (z v y))^y=y.
1435 [para_into,1320.1.1.2.1,572.1.1] x v (x^y)=x.
1441 [para_into,1320.1.1.2,590.1.1] x v ((y^x) v x)=x.
1483 [para_into,1395.1.1.1.2,364.1.1] ((x v y)^ (y v z))^y=y.
1644 [para_from,1435.1.1,500.1.1.2] (x^y)^x=x^y.
1799 [para_from,1441.1.1,500.1.1.2] ((x^y) v y)^y= (x^y) v y.
1909 [para_from,1644.1.1,1320.1.1.2.1.1] x v (((x^y) v x)^z)=x.
2405 [para_into,1799.1.1,700.1.1,flip.1] (x^y) v y=y.
2577 [para_into,1909.1.1.2,788.1.1] x v (y^ ((x^z) v x))=x.
2662 [para_into,2405.1.1.1,731.1.1] ((x^y) v (z^y)) v y=y.
2666 [para_into,2405.1.1.1,500.1.1] x v (y v x)=y v x.
2716 [para_into,2577.1.1.2,76.1.1] x v (y^ (x^z))=x.
2804 [para_into,2662.1.1.1.2,1644.1.1] ((x^y) v (y^z)) v y=y.
3025 [para_from,2666.1.1,265.1.1.2] x^ (y v (x v z))=x.

6.2. Classical Propositional Logic

For readers interested more in logic than in mathematics, I return to classical propositional calculus. Lukasiewicz gave the following single axiom for the implicational fragment of that area of logic; no shorter axiom exists.

P(i(i(i(x,y),z),i(i(z,x),i(u,x)))).

The following 3-basis, from Tarski-Bernays, provides an axiom for the implicational fragment.

P(i(x,i(y,x))).
P(i(i(i(x,y),x),x)).
P(i(i(x,y),i(i(y,z),i(x,z)))).

OQ18.PC: What is the length of the shortest proof (relying solely on condensed detachment) showing that the given Lukasiewicz 13-letter formula is a single axiom for the implicational fragment of two-valued sentential calculus? (On May 24, 2000, I discovered a 32-step proof, strictly shorter than any previously known; see Section 3 for the proof.)

Lukasiewicz, in the 1930s, presented the following 23-letter single axiom for all of classical propositional calculus.

P(i(i(i(x,y),i(i(i(n(z),n(u)),v),z)),i(w,i(i(z,x),i(u,x))))).

(Years later, Meredith found a 21-letter single axiom for this area of logic; it is strongly believed, but not yet proved, that no shorter single axiom exists.)

Q23.PCa: With or without requiring the avoidance of double negation, does there exist a proof shorter than length 50 relying solely on condensed detachment that shows the Lukasiewicz 23-letter formula to be a single axiom? (I have found a 50-step proof, the following, a proof free of double-negation terms; that proof deduces the very familiar 3-basis of Lukasiewicz.)

A 50-Step Proof for the Lukasiewicz 23-Letter Single Axiom

----- Otter 3.0.5b, March 1998 -----
The process was started by wos on soot.mcs.anl.gov, Thu Aug 2 19:56:49 2001
The command was "otter". The process ID is 3520.

-----> EMPTY CLAUSE at 0.49 sec ----> 138 [hyper,7,135,128,105] $ANS(step_allLuka_1_2_3).

Length of proof is 50. Level of proof is 36.

---------- PROOF ----------

1 [] -P(i(x,y))| -P(x)|P(y).
7 [] -P(i(i(p,q),i(i(q,r),i(p,r))))| -P(i(i(n(p),p),p))| -P(i(p,i(n(p),q)))|$ANS(step_allLuka_1_2_3).
8 [] P(i(i(i(x,y),i(i(i(n(z),n(u)),v),z)),i(w,i(i(z,x),i(u,x))))).
79 [hyper,1,8,8] P(i(x,i(i(i(i(y,z),i(u,z)),i(z,v)),i(w,i(z,v))))).
80 [hyper,1,79,79] P(i(i(i(i(x,y),i(z,y)),i(y,u)),i(v,i(y,u)))).
82 [hyper,1,8,80] P(i(x,i(i(i(y,z),i(i(u,y),i(v,y))),i(w,i(i(u,y),i(v,y)))))).
83 [hyper,1,80,8] P(i(x,i(y,i(i(y,z),i(u,z))))).
84 [hyper,1,82,82] P(i(i(i(x,y),i(i(z,x),i(u,x))),i(v,i(i(z,x),i(u,x))))).
85 [hyper,1,83,83] P(i(x,i(i(x,y),i(z,y)))).
86 [hyper,1,84,85] P(i(x,i(i(i(y,z),y),i(u,y)))).
87 [hyper,1,86,86] P(i(i(i(x,y),x),i(z,x))).
88 [hyper,1,8,87] P(i(x,i(i(y,i(y,z)),i(u,i(y,z))))).
89 [hyper,1,88,88] P(i(i(x,i(x,y)),i(z,i(x,y)))).
90 [hyper,1,89,89] P(i(x,i(i(y,i(y,z)),i(y,z)))).
91 [hyper,1,90,90] P(i(i(x,i(x,y)),i(x,y))).
92 [hyper,1,91,8] P(i(i(i(x,y),i(i(i(n(z),n(u)),v),z)),i(i(z,x),i(u,x)))).
93 [hyper,1,92,92] P(i(i(i(x,y),i(y,z)),i(u,i(y,z)))).
94 [hyper,1,92,91] P(i(i(x,i(i(n(x),n(y)),z)),i(y,i(i(n(x),n(y)),z)))).
95 [hyper,1,92,93] P(i(i(i(x,y),i(z,x)),i(u,i(z,x)))).
96 [hyper,1,91,93] P(i(i(i(x,y),i(y,z)),i(y,z))).
97 [hyper,1,91,95] P(i(i(i(x,y),i(z,x)),i(z,x))).
98 [hyper,1,96,8] P(i(i(i(i(n(x),n(y)),z),x),i(i(x,u),i(y,u)))).
99 [hyper,1,97,85] P(i(i(i(i(x,y),z),y),i(x,y))).
100 [hyper,1,98,96] P(i(i(i(n(x),y),z),i(x,z))).
101 [hyper,1,99,98] P(i(i(n(x),n(y)),i(i(x,z),i(y,z)))).
102 [hyper,1,99,95] P(i(x,i(y,i(z,x)))).
105 [hyper,1,100,91] P(i(x,i(n(x),y))).
107 [hyper,1,100,102] P(i(x,i(y,i(z,i(n(x),u))))).
108 [hyper,1,96,107] P(i(x,i(y,i(n(i(z,x)),u)))).

109 [hyper,1,94,108] P(i(x,i(i(n(y),n(x)),i(n(i(z,y)),u)))).
110 [hyper,1,109,94] P(i(i(n(x),n(i(i(y,i(i(n(y),n(z)),u)),i(z,i(i(n(y),n(z)),u))))),i(n(i(v,x)),w))).
111 [hyper,1,97,110] P(i(n(i(x,y)),n(y))).
112 [hyper,1,101,111] P(i(i(i(x,y),z),i(y,z))).
113 [hyper,1,112,8] P(i(i(i(i(n(x),n(y)),z),x),i(u,i(i(x,v),i(y,v))))).
114 [hyper,1,99,113] P(i(i(n(x),n(y)),i(z,i(i(x,u),i(y,u))))).
115 [hyper,1,92,113] P(i(i(i(i(x,y),i(z,y)),i(i(n(x),n(z)),u)),i(v,i(i(n(x),n(z)),u)))).
116 [hyper,1,92,114] P(i(i(i(i(x,y),i(z,y)),n(x)),i(u,n(x)))).
117 [hyper,1,115,89] P(i(x,i(i(n(y),n(i(y,z))),i(i(y,z),z)))).
118 [hyper,1,92,116] P(i(i(n(x),i(i(x,y),i(z,y))),i(u,i(i(x,y),i(z,y))))).
119 [hyper,1,117,117] P(i(i(n(x),n(i(x,y))),i(i(x,y),y))).
120 [hyper,1,91,118] P(i(i(n(x),i(i(x,y),i(z,y))),i(i(x,y),i(z,y)))).
121 [hyper,1,100,119] P(i(x,i(i(x,y),y))).
122 [hyper,1,85,121] P(i(i(i(x,i(i(x,y),y)),z),i(u,z))).
123 [hyper,1,115,122] P(i(x,i(i(n(y),n(z)),i(z,i(i(y,u),u))))).
124 [hyper,1,123,123] P(i(i(n(x),n(y)),i(y,i(i(x,z),z)))).
125 [hyper,1,100,124] P(i(x,i(y,i(i(x,z),z)))).
126 [hyper,1,120,125] P(i(i(x,y),i(i(n(x),y),y))).
127 [hyper,1,112,125] P(i(x,i(y,i(i(i(z,x),u),u)))).
128 [hyper,1,97,126] P(i(i(n(x),x),x)).
130 [hyper,1,120,127] P(i(i(x,y),i(i(i(z,n(x)),y),y))).
131 [hyper,1,130,130] P(i(i(i(x,n(i(y,z))),i(i(i(u,n(y)),z),z)),i(i(i(u,n(y)),z),z))).
133 [hyper,1,98,131] P(i(i(i(i(i(x,n(y)),z),z),u),i(i(y,z),u))).
135 [hyper,1,133,98] P(i(i(x,y),i(i(y,z),i(x,z)))).

Meredith supplied the following 21-letter single axiom for all of classical propositional calculus in terms of implication and negation.

P(i(i(i(i(i(x,y),i(n(z),n(u))),z),v),i(i(v,x),i(u,x)))).

OQ20.PC: With no constraint on the type of term present, what is the length of the shortest proof that relies solely on condensed detachment and on the Meredith single axiom and that completes with the deduction of some known basis for classical propositional calculus? (With OTTER, in September 2000, I discovered three proofs each of length 38, shorter by three steps than Meredith's 41-step proof; each of the proofs deduces the well-known Lukasiewicz 3-basis, and each relies heavily on double negation; see Section 4.2 for one of the proofs, an amazing proof.)

7. Commencing a Study of Hilbert's 24th Problem

Ever since Thiele first found the 24th problem in Hilbert's handwritten notes in the library of Goettingen, some mathematicians have posed questions focusing on what Hilbert meant and what is involved in solving this problem of proof simplification. Indeed, a quick search in Google turns up over 800 references to the 24th problem. Would Hilbert, if given the time to make precise the objective of the 24th problem, have discussed proof length, proof complexity, lemma avoidance, term avoidance, and such? These aspects are among those certainly relevant to simplicity of proof, aspects (featured here) that can be addressed with the use of a powerful reasoning program. In contrast, an aspect that presents more difficulty in addressing is proof size, the total number of symbols found among the deduced steps of a proof when taken together. Not directly, nevertheless, I have two proofs showing the formula *XCB* to be a single axiom for equivalential calculus, each deducing a 2-basis consisting of symmetry and transitivity, the first having size 686, and the second having size 772. Not only do both proofs share their key deductions, symmetry and transitivity, but both have length 22 (applications of condensed detachment.) No surprise, I am certain that Hilbert would have preferred the smaller in size. In addition, the smaller has variable richness 16 and complexity 63, whereas the larger in size has richness 19 and complexity 75.

My Hilbert approach to the problem has, of course, involved the use of automated reasoning. I conjecture that Hilbert himself would have applauded if presented with some of the proof refinements that have been obtained with an automated reasoning program, proofs that are sometimes simpler than those found by masters. Further, because of Hilbert's clear preference for rigor and formalism, I conjecture that he would give the highest grade to the proofs OTTER completes—proofs for which OTTER cites both the history and the inference rule used in each step. In this notebook, you learn of approaches that have proved successful in discovering simple proofs, simple in various respects. The results presented here, at least implicitly, offer challenges to be met and questions to be answered. In Sectionn 6, among other items, you are presented with specific open questions for study.

For those who enjoy history, if memory serves, my first study of proof shortening concerned the dependence of *MV*5 on the other four axioms of the 5-axiom system of Lukasiewicz for his infinite-valued sentential calculus. McCune and I first found a 63-step proof, and, eventually, when I turned to the goal of finding a short proof, I found one of length 37. That same theorem, from what I can recall, was also my first study focusing on the avoidance of double-negation terms, and I found a 37-step proof free of such terms. And, as you now know, after perhaps eight years of more than occasionally seeking so-called short proofs, how satisfying it was to learn that this aspect of proofs was pertinent to a problem posed by Hilbert.

Until the next notebook, assuming that such will be written, I invite you to devote some effort to various aspects of the Hilbert 24th problem. If you do so, even if your studies do not yield monumental results, you may experience some or much of the joy I have experienced working on this charming, tantalizing, and occasionally subtle problem.