NEGLECTED POSSIBILITIES OF PROCESSING ASSERTIONS AND PROOFS MECHANICALLY: CHOICE OF PROBLEMS AND DATA

by

GEORG KREISEL
Department of Philosophy
Stanford University

1. BACKGROUND AND PRINCIPAL CONCLUSIONS

FIFTY YEARS of research in logic have shown that most—of the logically—natural general problems cannot be solved mechanically. This research began in the thirties with the famous (recursive) incompleteness and undecidability results, and culminated in the seventies with large lower bounds for the number of steps needed to execute earlier "positive" results. Thus, at least as long as the traditional parameters for classifying problems are used, for example, number of symbols (of formulae or derivations in any of the usual formal systems), essentially all known proof and decision procedures were shown to grow too fast for realistic use. These facts are by now well known, but not the obvious proviso that the negative results are tied to classes of problems chosen in traditional logical terms. The neglected possibilities mentioned in the title involve more sophisticated choices, of problems and data, so to speak, between the traditions of mathematical logic and of other systematic expositions, familiar for example from Bourbaki. The principal conclusions, supported in detail below, may be summarized as follows.

1. Contrary to a widespread impression, the mechanical *checking* of (the validity of) theorems and informal proofs is much more closely related to experience in ordinary mathematics, with its stress on so-called impure proofs combining knowledge from different branches or "structures", than to experience from logic with its emphasis on universal systems. Furthermore, the mathematical emphasis affects not only methods of proof, but leads to sophisticated choices of problems, namely, those which permit interpretation in terms of the basic structures considered.
2. Contrary to most elementary texts on logic, the heart of logical studies of proofs does not concern the building up of formal derivations (by means of

formal rules), but with their transformations. This corresponds to the *unwinding of proofs,* familiar in mathematics from reading off bounds from proofs of existential theorems or in computer science from extracting "straight code", given a "high level" description of an algorithm (by means of a proof that the intended algorithm satisfies given specifications).
Obvious, but neglected, possibilities of using transformations of proofs include the mechanization of the two kinds of familiar applications just mentioned, especially if the number of cases or the length of the transformation is too great to be done by hand. A less obvious, but equally valuable use is in the *mechanical verification* (rather than: synthesis) *of programs,* in the sense that the program is not only shown to define an operation satisfying the given specifications, but that it implements the intended algorithm. A practically less important use of such transformations is in the *routine axiomatization* of (nonaxiomatic) proofs.

3. Concerning the choice of data for representing proofs, the aims of 2 are particularly useful because the transformations used are quite complicated, and so superficially small changes in the data can make a big difference to the execution. This sensitivity is of help in making a choice of data. In contrast, if only checks on validity are involved, as in 1, the relevant requirements of simplicity and reliability (of principles of proof) leave too much room for a convincing choice. Even more delicate is the choice of data if the latter are to make explicit the psychologically significant features of proofs (in the ordinary sense of this word, referring to mental phenomena), just because even very crude and "partial" data are sufficient to identify the object, that is, the proof in question. In medieval terminology, different descriptions which are sufficient to identify the object meant are called *extensionally equivalent,* and the description of the particular features that happen to have attracted our attention is called *intensional:* neither of these extremes can be expected to help in a theoretical study of the phenomena of proofs, let alone their efficient mechanization.

1.1 *A Problem of Perspective*

Readers should not be deceived by the fact that both the conclusions above and the technical knowledge used below to support the latter, are intrinsically quite easy. A very real difficulty remains inasmuch as the aims which originally led to the (technical) knowledge, both in logic and other mathematics, were quite different from those of the present article; in fact, in a sense made precise below, many of the old aims are in *demonstrable conflict* with the new ones. This conflict is of course consistent with the "negative" results mentioned at the beginning. As a matter of practical politics, constant attention to this conflict is needed, because the most elementary or "basic" strategies are affected by it, for example, the choice of relevant equivalence relations (between derivations). A more specific consequence of this conflict is sociological: the relevant parts of logic, especially proof theory, have not become widely known, but only those parts devoted to dubious foundational aims. As a result the whole subject is in bad odor, especially among perceptive scientists, that is, just the people who potentially are best fitted to advance the subject.

1.2 *A Reminder of the Simplest Type of Efficient Mechanization*

The best guarantee is to find programs which are not too hard to execute, but can be applied to very "many" instances of interest. The alternative, where very long computations are needed for a few answers, is liable to get out of hand, particularly in the present context since, again by the negative results mentioned, there is a relatively narrow range which is feasible for computers, but not for us. Experience suggests that, at least in pure mathematics, relatively few instances are already too "many", because the repetition of routine operations is boring and tiring, and thus liable to error.

1.3 *Terminology: Automatic Theorem-proving*

Though the techniques of checking and unwinding (proofs) are very different indeed, the results are all, in a quite natural sense, instances of automatic theorem-proving. This is clear for checking the validity of a theorem. But also in the case of unwinding (see sec. 3) of a derivation d of $\forall x\ \exists y\ R(x,y)$ and improving the algorithm α_d implicit in d, an automatic proof of the theorem $\forall x\ R[x, \bar{\alpha}_d\ (x)]$ is supplied where $\bar{\alpha}_d$ is the improved algorithm. The terminology: "unwinding" a given proof (instead of "finding" the unwound proof) is used to emphasize that the details of the given proof are used, and not only the form of the end formula of the unwound proof.

2. LESSONS FROM PURE MATHEMATICS: CHECKING THEOREMS AND PROOFS

The particular strategy of organizing an area of knowledge, which serves us here as a model, is the style of Bourbaki: one looks for a *few* definitions and key theorems that lead to easy solutions of *many* problems. (No one proof in Bourbaki is long.) This is both in contrast to using a single primitive, as in set theory, and to using a large number of diverse methods as in traditional number theory; we shall return to more modern sophisticated choices of number-theoretic problems in a moment. When we come to mechanizing (our knowledge of) an area of knowledge, the discovery corresponding to those "few" definitions and theorems is the human part of the enterprise, while the "easy" solutions are to be mechanized.

The single most neglected aspect—in the logical, not the mathematical literature on systematic organization—is the role of *impure methods* in the human part; examples are geometric methods in algebra, algebraic and topological ones in number theory, and in fact combinatorial methods in logic itself. This is in sharp contrast to both logical ideals (a number-theoretic theorem should have a number-theoretic proof) and to most heavily advertised results of logic, such as completeness theorems which establish the theoretical possibility of proving every theorem of the system

considered by pure methods, for example, logical theorems by purely logical methods.

2.1 *The Horror of Impure Methods: Real and Imagined*

If mechanization is contemplated at all, an unrestricted use of impure methods would indeed be disastrous. We should be back to Hegel's business about everything being connected with everything else! Also, even if a limited number of impure methods, for example, Bourbaki's (few) so-called basic structures with their principal properties, are found to be effective for organizing a given area of knowledge, it will rarely be possible to give convincing theoretical reasons for the success of the particular choice of methods, except when the subject is highly developed. This is in sharp contrast to the immediate appeal—especially to the outsider!—of such logical ideals as completeness. Indeed, the success itself depends on a shift of interest in the light of experience, as illustrated by the following standard example.

2.2 *Diophantine Equations: From Classifications by Degree and Number of Variables to Geometric Properties of Varieties Over Finite Fields*

Long before the recursive undecidability of Hilbert's tenth problem for diophantine equations (in integers) was established, number-theorists introduced quite sophisticated classifications of equations $P = 0$, by the *genus* of the variety defined by $P = 0$ (when the variables range over suitable fields). A first indication that this choice was fruitful comes from Siegel's theorem for curves, that is, equations $P = 0$ in two variables, which turn out to have only finitely many integral zeros if the curves are of genus > 0. (For the cases of genus 1 and 2, Hilbert's problem is decidable.) Certainly, naively, one is tempted to object: What have geometric properties of genus to do with number theory? But it is also evident that the negative results stated at the beginning of this article do not apply even asymptotically to such radically different classifications.

2.3 *General Strategy*

There is, obviously, no full substitute for detailed experience of the kind of organization involved. But it may be of use to (some) readers that the general scheme can be illustrated by examples from very diverse branches of mathematics, even propositional logic. Here the guiding principle is implicit in the familiar "reduction" of finite problems to propositional logic but with a reversal of sign! Instead of thinking of propositional logic as a *foundation* for (branches of) finite mathematics, one uses knowledge about the latter to prove propositional theorems by inspection. For example, suppose the formula P_{MN} expresses the instance of the pigeonhole principle: if $N > M$ and N objects are placed in M pigeonholes, at least one pigeonhole contains more than one object. For large N and M, the verifica-

tion that P_{MN} is an instance of the pigeonhold principle (according to the natural formal scheme) is simpler than, say, a formal derivation of P_{MN} in one of the usual systems of propositional logic. Clearly, that verification can be mechanized too.

2.4 *Remark on Proof-Checking*

Just as a body of knowledge can be organized, so individual proofs can be organized by picking out key definitions and lemmas in such a way that the verification of the relevant implications is almost instantaneous, while it would be hard to find a "missing" lemma; cf. the difference between proof search and proof-checking. *One* possibility of exploiting the empirical fact that we learn to organize proofs or, more precisely, proofs occurring to us after a certain drill in mathematics (not: all valid proofs) is this: there is a relatively *simple decision procedure for the implications* which we tend to leave implicit. This would be parallel to Wang's discovery some 25 years ago that the theorems of *Principia,* which Whitehead andRussell listed as logical, that is, tacitly: as typical of logic, were all easily transformed into formulae of a decidable subclass. In other words, in the traditional (ordinary) sense of elementary logic, the list in *Principia* was not typical at all, validity for elementary logic being recursively undecidable. It will not have escaped the reader's attention that, once again, we have a conflict between the traditional logical classifications and those suited for mechanical processing (of assertions, represented by formulae): otherwise we should have to worry about the negative results at the beginning of this article.

It should be mentioned that the term "proof-checking" used above with a view to mechanization is used in quite different ways when applied to ordinary human reasoning. First of all, strictly speaking, the mechanization envisaged is primarily intended to check the validity of the final theorem: the given proof and its elaboration by mechanical means are tools towards checking the validity of the end formula. This corresponds to checking the *results* of a computation, not primarily whether the computation contains a gap (according to some more or less arbitrary list of formal rules). But in human reasoning we normally check the validity of theorems or computations not primarily by filling gaps or retracing earlier steps but by use of independent information, for example, by applying commutativity to compare $a + b$ and $b + a$.

2.5 *Disclaimer*

In contrast to the next section, on the unwinding of proofs, I myself have not made any detailed studies on the topics touched above. I am not at ease with the type of empirical results (on "easy" implications) conjectured here, although I believe them to be useful. They bring to mind such geophysical discoveries as: on the west coast of France every seventh wave tends to be high, on the Pacific Coast of the United States it is every ninth wave. Such information is useful, but it can hardly be claimed that the

theoretical explanation—in terms of the distance and area of the storm centers where the waves are generated, and form wave trains by interference—is particularly inspiring.

3. UNWINDING OF PROOFS: SYNTHESIS AND VERIFICATION OF PROGRAMS

This section contains the—so far—most successful uses of mathematical logic, in particular, proof theory for mechanizing (routine) reasoning. The reasons for the success are apparent from the following historical review.

3.1 *Three Stages of Unwinding*

The first stage goes back to the literature on so-called nonconstructive existence proofs. The latter have been hotly debated for nearly 100 years: criticized for "hiding" explicit realizations in indirect *ad absurdum* arguments (or, for that matter, in logically complex so-called intuitionistic arguments) which can be difficult to unwind. The first concern was to show that nontrivial realizations can be extracted at all, from proofs of so-called $\forall\exists$ theorems generally, and from arbitrary intuitionistic proofs of existential theorems (where, in comparable classical and intuitionistic systems the processes of unwinding required are of about the same computational difficulty).[1] In the thirties and forties various transformations of proofs, known as cut-elimination and later as normalization, and various functional interpretations were developed to carry out the unwinding. This aim replaced the earlier consistency program of Hilbert, which differed from unwinding in two principal respects:

1. Only purely universal, not existential statements were considered.
2. The main emphasis was on the (elementary) proofs used, not on the operations used in unwinding.

For the latter, it is immaterial whether the proof establishing the *correctness* of the unwinding procedure is or is not elementary (provided the proof is sound). Actually, the unwinding involved could be read off from earlier consistency proofs: this possibility had been neglected in accordance with then-current foundational preoccupations with restrictions on metamathematical methods to be used in consistency proofs—preoccupations which had caught the imagination despite the fact that it was not very clear what more we knew from an elementary proof of a universal proposition, for

[1]For specialists: the corresponding formal fact is the easy proof of closure under Markov's rule. One speaks of "nontrivial" realizations in the case of $\forall\exists$ theorems, say $\forall n\ \exists m\ R(n,m)$ since, if such a theorem is true at all, a "trivial" realization is found by trial and error: try out $R(n,0), R(n,1), \ldots$ to get the least m_n: $R(n,m_n)$.

example, of an identity than from a civilized proof. (In contrast, some civilized proofs of existential theorems tended to be more difficult to unwind.) During the first stage just described there were some successes, in particular the unwinding of Littlewood's proof or, more precisely, of two quite different proofs (using respectively the Riemann hypothesis and its negation) on the oscillation of $\pi(x) - li(x)$ or of Artin's theorem on sums of squares (without the need for new ideas).

The second stage of work on unwinding, beginning in the late sixties, concentrated on *comparing* the results of different unwinding processes. The upshot was that, with suitable provisos, different processes μ applied to the same formal derivation d of an $\forall\exists$ theorem defined the same realization μ_d up to a natural equivalence or convertibility relation. In other words, the idea of unwinding, that is, of extracting *the* algorithm provided by a given derivation d, was formally "legitimized" *modulo* the equivalence relation in question. As an obvious corollary, one had a natural (precise) meaning both for mechanical synthesis and mechanical verification of programs, to which we return in a moment. Though these ideas date back to the early seventies (cf. Kreisel, 1971), they were published more fully in lectures at Clermont-Ferand in 1975 (Kreisel, 1977b) and Belgrade in 1977 (Kreisel, 1977a).

At the third stage of work on unwinding, the question (asked by Goad, 1980) was: *Can't we do better?* Specifically, instead of simply extracting the algorithm μ_d provided by a given derivation d, are there not mechanical methods for *improving the efficiency* of μ_d? (In computer jargon, "optimize" is used for "improve".) Thus, far from merely wishing to legitimize the idea of a specific algorithm α_d being implicit in a derivation d, of say $\forall x\,\exists y\,R(x,y)$, Goad looked for an extensionally different algorithm $\bar{\alpha}_d$, obtained mechanically from α_d, which also satisfied $\forall x\ T[x,\bar{\alpha}_d(x)]$. He found, in his dissertation, convincing examples where α_d contained *latent redundancies*, which can be removed mechanically by *modifying* existing normalization procedures or, equivalently, functional interpretations; "latent" because the redundancies do not affect the algorithm α_d itself, but only its application to specific arguments. (For example, $(a-b)c$ cannot be simplified as a function of the three variables a, b, c; but it can, if $b = a$ and a function of a and c alone is required.) Readers are referred to Goad's dissertation (1980) for further details.

3.2 *Rewording in Computer Terminology*

A proof, represented by a formal derivation d of an $\forall\exists$ theorem, may be regarded as a *high-level description* of an algorithm (which defines a function that satisfies the given $\forall\exists$ theorem). Unwinding the proof corresponds to *extracting straight code* μ_d, from the derivation d. Roughly speaking, the more abstract the principles formalized in d, the higher the level of the descrip-

tion; more precisely, the more abstract the principles the greater is the potential simplification (of d compared to μ_d). Obviously, not only simplification, but also other differences can be of value, for example, familiarity of d, again compared to μ_d; some discretion is needed in applying the requirement of familiarity or closeness to ordinary language since it would be unreasonable to try to perpetuate defects of the current state of knowledge.

Extracting straight code: $d \mapsto \mu_d$, may also be described as the synthesis of a program (μ_d). It is mechanical, if, as in the usual proof theoretic literature on normalization and functional interpretation, the map μ is recursive. If it is easy to verify, by inspection, that d correctly formalizes a given proof (of the $\forall\exists$ theorem considered), the *mechanical synthesis of the program* μ_d serves the purpose of mechanically *verifying that* μ_d not only defines a realization of the given $\forall\exists$ theorem but actually *implements the intended algorithm.* A more legalistic but less useful definition of *verifying a program* μ' is that μ' and μ_d are (mechanically) interconvertible by appropriate rules. This relation is mechanically decidable, ensures that μ' and μ_d define the same function, but not necessarily that μ' implements the intended algorithm.

Evidently, there is a separate—and prima facie more economical—problem: to verify that μ' defines *some* function which solves the $\forall\exists$ problem considered. But this problem is, as is well known, not generally mechanically solvable. However, as is also well known from general scientific experience, it is by no means unusual that a more informative or complete solution is actually easier, that is, more economical: the possibility of an (easy) informative solution is the result of a clearer understanding of the "epistemological" situation, in the present case, of understanding the implications of having the derivation d at hand.

3.3 *Neglected Uses of Mechanical Unwinding in Pure Mathematics*

While, as mentioned already, good uses of unwinding (by hand!) were made around 25 or 30 years ago, between 1956 and 1976 very little new material appeared in the mathematical literature which lent itself to unwinding. Since then, very striking proofs of $\forall\exists$ theorems have been discovered which raise obvious problems of unwinding, possibly solvable by mechanical means (for the general reason already touched on in sec. 1.1). More specifically, a single abstract or "infinitistic" theorem I is proved from which a fairly large number of $\forall\exists$ corollaries $C_i (1 \leqslant i \leqslant N)$ are derived by a uniform scheme, for example, a compactness argument. A *single* program applied to derivations d_i (of C_i *via* I) then delivers unwound proofs of C_i with corresponding algorithms. *Examples* are

1. proofs of various finite versions of Ramsey's theorem I and
2. proofs of various n-dimensional versions of van der Waerden's or Szemeredi's theorems on arithmetic progressions from Fürstenberg's existence theorem I for idempotent elements.

For specialists: a little reflection shows that in case 1 there is a description of proofs in arithmetic for n-tuples ($n = 1, 2, \ldots$) by using a proof in arithmetic of the infinite version of Ramsey's theorem for each fixed n, and so here the proof theory of first-order arithmetic is appropriate, while in case 2 Fürstenberg's proof, by use of the theorem of Cantor-Bendixson, is most conveniently formalized in the theory of (generalized) inductive definitions. Since answers for several values of n are of interest, in case 2, to physicists working on percolation problems, it is reasonable to suppose that programs for mechanizing the unwinding in cases 1 and 2 are rewarding.

The list of candidates could be extended: for example, there are proofs of $L(1) \neq 0$ for various L-functions of (analytic) number theory which do not supply obvious lower bounds for $L(1)$, and are therefore also reasonable candidates for unwinding.[2]

3.4 *Routine Axiomatization or Algebraization*

Another use of unwinding comes from the use of *transfer principles* for proving theorems about axiomatically defined structures, for example: for every division algebra of dimension n over a real closed field, $n = 1, 2, 4, 8$. To show that there is no such algebra for, say, $n = 16$, it is sufficient to show that there is none over the field R of real numbers. This is done by (abstract)

---

[2]An elementary reference is Ellison (1975). Interested readers should work through the argument on pages 240–241 (for real characters χ) which derives a contradiction from (i) $L(1, \chi) = 0$ and Théorème 6.1 on pages 192–194 for Dirichlet series with positive coefficients, actually with $F(b) > F(a)$ in place of 6.1(b) on page 192. By (i), we have (ii) analytic continuation of $g(s)$ into $Re(s) \geq 1/2$ where $g(s)$ is

$$\zeta(s)L(s, \chi)/\zeta(2s),$$

and so

$$g(s) = \Sigma b_n n^{-s} \quad (b_1 = 1,\ b_n \geq 0)$$

for $Re(s) > 1$.

(a) A direct application of either cut elimination or functional interpretations (for arithmetic, in which the proof can be formalized) is long because the logically complicated assertion (ii) has to be analyzed. This is avoided by the following *mathematically trivial* switch to:

$$\text{(b) } G(s) = g(s) - \zeta(s)[L(1, \chi)/\zeta(2)],$$

where

$$G(s) = \Sigma[b_n - L(1, \chi)/\zeta(2)]n^{-s}.$$

$G(s)$ can certainly be continued to any point on (a strip around): $1/2 < s < 1$. This part of the argument need not be touched to get a positive lower bound for $L(1, \chi)$. Théorème 6.1, actually with a single circle Γ with center $(2, 0)$ and radius 1 1/2 (instead of $\Gamma_0, \ldots, \Gamma_k$ on page 193) shows: *not all* $b_n - [L(1, \chi)/\zeta(2)]$ *can be positive.* Thus there is a cut-free proof of $\forall n\ [L(1, \chi)/\zeta(2) > 0]$, which gives a rational $c > 0$: $L(1, \chi)/\zeta(2) > c$. (The dependence of c on the location of zeros of the ζ-function is easily written down.) *Open problem:* Find an analogue to the switch from (a) to (b) in the proof of 5.1 on pages 11–12 of Jacquet and Shalika (1976).

K-theory. From this follows the purely existential (Σ_1^0) statement expressing that some finite subset of the axioms for real closed fields implies logically that there is no division algebra of dimension $n = 16$. Such a finite subset (and the logical derivation in question) could be found by unwinding the proof by K-theory together with the transfer principle. The result can be compared to bounds obtained by quantifier-elimination. (Another problem in the same style, going back to Newton and Gregory, concerns the number of points, at linear distance $\geqslant 1$ on the unit sphere, namely $\leqslant 12$.)

3.5 *Open Problem on Unwinding*

Given a not obviously constructive proof of an existential theorem, to decide *whether or not* a realization in appropriate terms can be extracted either by straight unwinding or by adding some general background information.

3.6 *Diverse Remarks on Neglected Aspects*

First and foremost, and contrary to an almost universal misconception, it cannot be assumed that the unwinding of the proofs of C_i via the (strong) principle I will give approximately the same (bad) bounds for each $i \leqslant N$, namely a bound dominating all functions which can be proved to be recursive by means of I; for example, in case 1 of Ramsey's theorem, ϵ_0-recursive functions. What is true is that, if one knew nothing else about these proofs of C_i except that the principle I has been used, only such "worst possible" bounds could be asserted. But the unwinding process is complicated, and small changes in the data, that is, small differences between the proofs of C_i for different i, can make big differences in the results. There may be cancellations, latent redundancies, and so forth.[3] Indeed, it seems likely that a good deal of computational work will be needed before even remotely significant *detailed conjectures* about (cancellations in) the unwinding process can be stated—perhaps to be compared to striking numerical evidence which has changed number theory out of all recognition by such contributions as the conjecture of Birch and Swinnerton-Dyer. Secondly, and this conflicts with some of the more thoughtless exaggerations in the literature, efficient unwinding requires that the algorithmic content of a proof be separated from those parts which establish that the

[3]An admittedly unrealistic, but probably instructive example is obtained from formulae $\forall n[f(n) = 0]$ of the kind familiar from (Gödel's) incompleteness theorems. Suppose a "strong" principle I is needed to prove $\forall n[f(n) = 0]$, where f has a complicated definition. Now, of course $\forall n\, \exists m[f(n) = m]$ is a logical identity; but if it happens to be inferred from $\forall n[f(n) = 0]$, the unwinding of this derivation of $\forall n\, \exists m[f(n) = m]$ will not yield the trivial algorithm: $n \mapsto f(n)$, but the efficient algorithm: $n \mapsto 0$. In short, one need not be frightened of finding complicated algorithms merely because a strong principle I happens to be mentioned in a proof: in the case above, I is needed to *simplify* the algorithm.

algorithms involved satisfy certain desired so-called negative, in particular, universal conditions. Otherwise, the unwinding becomes inefficient, since the unwinding of those parts of the proof which lead to, say, universal conclusions does not generally contribute algorithmically useful information. (As a corollary, the algorithmic part is *not* the full content of a proof!) It is a separate question, incidentally of pure mathematical interest, what is gained by putting proofs of universal propositions into "normal form", for example, by use of suitably restricted transfinite induction. Lastly, in connection with the principle just mentioned, including definition by transfinite recursion, two quite separate elements must be distinguished: one is the (well-founded) ordering $<$ used, the other is the so-called descent function τ (where $A(x)$ is derived from

$$([\tau\ (x) < x] \rightarrow A[\tau\ (x)]) \rightarrow A(x)$$

and thus $A(x)$ holds if $\neg[\tau\ (x) < x]$). The distinction is particularly relevant for a comparison between recent work by Girard on infinite proofs satisfying strong uniformity conditions and earlier work following Gentzen. At the price of using orderings with larger ordinal, Girard can do with very simple descent functions. The latter seems important for computational efficiency, particularly when the arguments are small enough to be of practical interest, but this impression remains to be checked. Computers are probably essential for this purpose.

Readers familiar with the foundational literature should remember the warning, given earlier on, about the *conflict* between traditional foundational aims and those appropriate for computational purposes. Each of the remarks in the preceding paragraph illustrates one such conflict or another.

3.7 *Remark on More Recent Literature on Length of Procedures*

The discussion above, concerned with quite special classes of problems selected for their interest, is totally unaffected by exercises on *worst possible* or even *average case* analyses – tacitly, in a "logically" selected class of cases – including the problem whether $P = NP$. However appealing many of these exercises are mathematically, their practical value is small for reasons already stated at the beginning of this article.

4. CHOICE OF DATA: A RAPID SURVEY

The data considered are intended to represent proofs. The purpose of this section is to convey some idea of successful choices of data (also in other areas of science and technology), in particular, of the considerations which have helped to find such choices. Naturally, most of the material is intended for mechanical checking and unwinding of proofs; but for contrast,

and perhaps also for its intrinsic interest, there is a concluding paragraph on the choice of data for representing the mental phenomena of proofs.

It is a commonplace that the proper choice of data depends on what we want to "do" with them (and, tacitly, a small stock of data should serve many purposes). In more technical language, the choice of data depends on the—kind of—*transformations* to be applied to them.

4.1 *Examples*

A very simple, but exceptionally instructive, example is provided by (Goad's) pruning operations introduced at the third stage of unwinding of proofs. Here it is quite essential to introduce *explicitly* into the representation of proofs a device which marks the relevant case in an argument by cases or, in technical language, in $\vee$-elimination; cf. appendix 1 of Goad's dissertation (1980). This should be compared with the lecture, given by P. Martin-Löf at Orléans in 1972, where a corresponding device is introduced too, but, in contrast to Goad's dissertation, no specific significance or use of it is hinted at, let alone demonstrated. More elaborate examples come from pure mathematics.

Polynomial equations illustrate both the possibilities afforded if more knowledge (about the coefficients), so to speak enriched data are available, and the new possibilities of solving equations if the demands made on a solution can be reduced.

1. $x^2 - a = 0$: *real* solutions. Evidently, a must be given in such a way that $a \geqslant 0$ can be verified (if $a \geqslant 0$ is true). This is certainly so if a is given by a defining equation with integral coefficients for a (real) algebraic number. It would not be enough if a is given by a, say, Kalmar elementary rule for computing a decimal or oscillating decimal for a (even if a happens to be algebraic).

2. $x^{2n+1} + a_1x^{2n} + \ldots + a_{2n+1} = 0$: *real* solutions (for real coefficients a). Now zeros x can be computed from $(a_1, \ldots, a_{2n+1})$, but, in general, approximations to the a for the usual order or interval topology do not determine approximations to the set of zeros since small changes in the coefficients may alter the number of zeros. However, if the coefficients are given by oscillating decimals, it is possible to compute an approximation to *some* zero x from (sufficiently many) digits of the oscillating decimals for the coefficients, with one proviso: the zero may depend on the choice among the different decimals representing the same coefficients.

3. $z^n + a_1z^{n-1} + \ldots + a_n = 0$: *complex* solutions. Here there are many possibilities. One of the less well-known choices of data, also considered by Specker, maps n-tuples $(a_1, \ldots, a_n$ into n-tuples of not necessarily distinct zeros $(z_1, \ldots, z_n)$ with the natural notion of neighborhood: the distance between two sets of circles c and c' with centers $(a_1, \ldots, a_n)$, respectively

$(a_1', \ldots, a_n')$ is the *minimum* distance between the unions of c and c', respectively. For this topology one has a continuous mapping of coefficients into the set of zeros, multiple zeros being counted with proper multiplicity.

Readers will notice that the "general" or "systematic" language of topology has been used here. But it cannot be claimed that it adds much even for "orientation". The heart of the problem is the choice of the appropriate topology.

Continuous functions with real arguments and values illustrate the advantages of additional knowledge which were discovered not, so to speak, by general reflection but by looking at a common style of proof, namely, bisection arguments. For example, if $f(0) > 0$ and $f(1) < 0$, one finds a zero ξ of f in (0,1) by bisecting, seeing if $f(1/2) = 0$, > 0, or < 0, and putting $\xi = 1/2$ or else considering (1/2, 1), respectively (0, 1/2), and repeating the bisection. Now, in general, one cannot decide whether or not $f(1/2) = 0$. Clearly, the procedure can be continued, if for each $\epsilon > 0$, there is a ξ': $| \xi' - 1/2 | < \epsilon$ where $f(\xi') > 0$ or $f(\xi') < 0$, and so on for each bisection step. It turns out that many continuous functions of use in practice have this additional property, for example, analytic functions (which have a bounded number of zeros).

For reference below, readers should note that standard mathematical texts do not use such phrases as "advantages of additional knowledge", but rather talk of solutions in the category of mappings continuous for the topology of uniform convergence of the triple consisting of the continuous function f, a modulus of uniform continuity, and a mapping ξ', where $| \xi'(a, \epsilon) - a | < \epsilon$, and $f(\xi') > 0$ or $f(\xi') < 0$; the solution involved concerns the problem of mapping $\{f: f(0) > 0$ and $f(1) < 0\}$ into a zero of f.

Evidently the list of examples of including useful additional knowledge can be continued indefinitely. Here are two obvious, so to speak, extreme cases.

1. Inversion of $n \times n$ matrices is notoriously difficult. But if, for example, for theoretical reasons a_{ij} may be expected to be 0 for $i > j$, in other words, if the matrix is triangular, inversion becomes much simpler: so it is worth including: $a_{ij} = 0$ for $n \geqslant i > j$, as part of the data.

2. Another well-known example is provided by the integral $(2\pi i)^{-1} \int f'/f\, dz$ round a closed curve C in the complex plane when f is a regular function, which is bounded away from 0 on C. The value is integral, and so a rough calculation is enough to get a precise value. (Readers may like to formulate this result in terms of an appropriate topology on the space of regular functions f and on the map:

$$f \to ((2\pi i)^{-1} \int f'/f\, dz.)$$

4.2 *General Comments*

As is apparent from the second example above, readers are warned against the (naive) assumption that strange words mean unfamiliar concepts. Thus where we ordinarily speak of encoding additional knowledge which is—tacitly—useful and available, the mathematical literature will speak of "enriched structures". Similarly, the latter speaks of "continuity for a (suitable) given topology" where ordinarily one simply wants to calculate approximations from approximations.[4] Less obviously, but in accordance with the remark at the end of the first example above, though the mathematical literature contains generalities about enrichments and topologies, it does not set out reasons for the particular *choices* made: instead it "justifies" them by results. This presupposes of course that we are able to recognize the interest and relevance of the results (after an adequate amount of scientific experience), perhaps more reliably than we can judge explicit reasons for the choices. The role of background knowledge needed for sound choices in the mathematical and other sciences is often underestimated, especially in the hackneyed distinction between science and art, to be found in the popular literature on computing (a distinction which, as just mentioned, overestimates the systematic side of science and underestimates the role of guiding general ideas in art). A much more relevant distinction is that between science and technology, where the former studies the phenomena of nature which present themselves to us, while the latter solves, by any means we can get, tasks that we set ourselves.

Evidently, in most of the examples above, the choice of data is determined by transformations, for example, solutions of functional equations (transforming data for the parameters into data for the solutions). Sometimes, the choice of data is less sophisticated: if we happen to be able to decide, for some real number ρ and any rational r whether or not $r < \rho$, it will be sensible to use Dedekind cut representations for ρ even tbough we may not know how to add and multiply (decidable) cuts effectively.

4.3 *Remark on Data for Proofs in the Ordinary Sense: A Contrast*

Contrary to a widespread impression, there is no evidence at all that formal derivations according to the usual rules are as good, let alone better

[4]It should be observed in passing that, when only approximations, say, to real numbers are available, then a solution with good continuity properties will make for efficiency. To take a recent example (on completing the square), the representation

$$ax^2 + 2bxy + cy^2 = a(x + (b/a)y)^2 + (c - (b^2/a)y^2$$

is not good since b/a is not continuous near $a = b = 0$, but

$$\frac{(ax + by)^2 + (bx + cy)^2 + (ac - b^2)(x^2 + y^2)}{a + c} \text{ is.}$$

data for representing proofs than some "informal" way, such as the oral—not the ritualistic written—tradition of Bourbaki. As a matter of empirical fact, one follows and grasps a proof more reliably when it is presented informally. (As already indicated, not only the further step to a formal derivation in the mathematical sense, but also the step to "beautiful" mathematical prose is liable to be retrograde, to *spoil* understanding.) Though the formal representation is more "detailed" as one says, it is not adequate because it does not tell us explicitly how to *recover* the parts which are significant for understanding. This is particularly clear, when one is tired and one's memory does not function well: the formal derivation does not encode the links which correspond to the *memory structures* involved. So-called informal proofs, especially when supplemented with suitable drawings, are noticeably better. Another good test for the weakness of formal derivations as representations of thought is the subjective effort (boredom) involved in writing down such derivations: one violently overestimates the length of the derivation even after having executed it! None of these facts spoils the theme of the present article since the aims of the kind of mechanization pursued here do not depend on faithfulness to the processes of (human) reasoning; on the contrary, we have tried to mechanize those processes which we do badly or reluctantly.

4.4 *Disclaimer*

Obviously, the broad defects of (known) formal derivations as data for proofs do not exclude their *occasional uses* for answering particular questions about mental phenomena of reasoning. As an obvious example, formal results explain the difficulty of finding a proof if the theorem in question is independent of currently formulated axioms or requires, demonstrably, the introduction of auxiliary concepts. This is, perhaps, best compared to the *occasional use* of commonplace physical or geometrical principles in d'Arcy Thompson's *Growth and Form* to explain some isolated biological phenomena without introducing even a hint of *specifically* biological knowledge, say on the moledular level. (The formal results correspond to commonplace physico-geometrical principles, the specifically biological aspects to the mental activity involved in proofs). Certainly, there are surely plenty of phenomena which remain to be explained in a rough sort of way in the style of d'Arcy Thompson or by simple enrichments of current logical theory; cf. chapter 1 of Statman's dissertation on the use of *genus* (of derivations in the style of natural deduction) to analyze the effect of choosing explicit definitions and their properties skillfully. Perhaps the single most significant obstacle to applying experience in formal logic and mechanical processing to reasoning is that what *can* be (formally) built from given operations, is *not* necessarily so built up, but may be realized independently, so to speak, redundantly: the discovery of these "bricks" in (human) information processing, of special "chips" in the nervous system,

is of the essence. Thus there are obvious redundancies, including relics of earlier stages of evolution and all the rest, which go counter to the most elementary ideas of "elegance" and "plausibility". This is another reason why, as stressed throughout this article, we do not rely at all on detailed experience with human reasoning in our choice of aims and methods for mechanization.

4.5 *Literature*

It has to be admitted that this article might have been more useful if all *obiter dicta* had been supported by reference to technical results in the literature. Interested readers are advised to look at Goad's dissertation (1980), and pursue his bibliography. A convenient source of references to the problem in pure mathematics mentioned in section 2 is the article by Erdös and Graham (1979).[5]

REFERENCES

Ellison, W. J. *Les nombres premiers*. Paris: Hermann, 1975.

Erdös, P., & Graham, R. L. Old and new problems and results in combinatorial number theory: van der Waerden's theorem and related topics. *Enseignement Mathématique,* **25,** 1979, 325–344.

Goad, C. A. *Computational uses of the manipulation of formal proofs* (Computer Science Dept. Rep. STAN-CS-80-819). Stanford, Calif.: Stanford University, 1980.

Jacquet, H., & Shalika, J. A. A non-vanishing theorem for Zeta functions of GL_n. *Inventiones mathematicae,* 1976, **38,** 1–16.

Kreisel, G. Hilbert's programme and the search for automatic proof procedures. *Zentralblatt für Mathematik*, 1971, **206,** 277–278.

Kreisel, G. From foundations to science: Justifying and unwinding proofs. *Recueil des travaux de l'Institut Mathématique, Belgrade* (Nouvelle serie), 1977, **2,** 63–72. (a)

Kreisel, G. Some uses of proof theory for finding computer programs. In M. Guillaume (Ed.), *Logique: Colloques internationaux du CNRS*. Paris: Editions du CNRS, 1977, (b)

---

[5]There is an apparently innocuous switch on page 327 from "does not give any usable bounds" to "furnishes no estimates" (in lines 14–15, resp. line 17) in connection with (a) Szemeredi's, resp. (b) Fürstenberg's proof. However, this switch may correspond to the following precise difference. In terms of Ackerman's function enumerating the primitive recursive functions, bounds can actually be read off by hand from (a); but they are not "usable" because their size – not their definition! – is huge. Though, for example, cut-elimination for the theory of generalized inductive definitions, in which (b) can be formalized, provides a theoretical procedure for defining an estimate (of $r_k(n)$ involved) in terms of a formalized proof, the procedure itself may not be feasible: so even the definition, quite independently of the size of $r_k(n)$ may be unusable.

Contrary to the impression conveyed by page 327 of Erdös and Graham (1979), not all the impure (non-combinatorial) proofs of van der Waerden's theorem are hard to unwind. Specifically, the iteration of the fixed point construction (Prop. 1.2 on page 65) in the proof of

Fürstenberg and Weiss by *topological dynamics* yields, once again, bounds of the order of Ackermann's function. Readers are warned that this fact provides little evidence for supposing that Ackermann's function is close to optimal bounds for van der Waerden's theorem itself; at best, there is a *generalization* of that theorem which can be proved by the methods involved and has optimal bounds of the order of Ackermann's function.