Logique & Analyse **127-128** (1989), 337-363

NATURAL HEURISTICS FOR PROOF CONSTRUCTION. PART I: CLASSICAL PROPOSITIONAL LOGIC.

Diderik BATENS

1. *Introduction*

Standard formal systems, whether axiomatic, in Gentzen style, or other, contain a definition of "a proof of B from A_1, ..., A_n" and usually an algorithm for recognizing proofs may easily be devised. Neither of these provides us with a way to find proofs.

As in other domains, we have witnessed a tendency to not only concentrate on results (in our case proofs) but on processes (in our case heuristics for proofs) as well. Studies on algorithms for proof construction in (fragments of) formal systems have ultimately led to automatic theorem proving and its impressive instruments. The latter are efficient and fast, but they are not *natural* in the sense that (most) humans (most of the time) search for proofs along different lines.

In this paper I present a first set of results on natural heuristic methods for constructing proofs. Given a set of rules of inference and a format for writing proofs(1), a heuristic method is a *set of instructions* that have to be executed in some *order*. The instructions may both refer to the items that constitute the proof and to other aspects of the format of proofs; most importantly, they may refer to the overall or local *goal* — see section 2. If a heuristic method simply acts on all separate items in the proof (in some order), it will be called non-contextual; a *contextual* method reacts differently to an item according to the *stage* of the proof (everything actually written down at some point in time). The distinction may be illustrated by tableau methods. These are usually non-contextual: once the initial formulas have been written down, a tableau method may simply run through all formulas and apply the appropriate rule to each of them. A contextual heuristic method may, in view of other available formulas, select the formula to be operated upon first, react differently to a formula, etc.

(1) E.g., the fact that each line of the proof contains a line number, the way in which the justification of a formula is worded, etc.

Natural heuristic methods should (i) be goal-oriented, (ii) be algorithmic whenever possible, (iii) embody insights on the search for proofs, and (iv) allow being extended in such a way that more insights are built in (thus leading to more 'intelligent' proofs).

Some things should be clarified right away. By a natural heuristic method for proof construction I mean one that agrees, in an explicatum-like manner, with the way in which people search for proofs when doing so unconsciously. I do by no means believe that there are innate heuristic methods for reasoning. (I conjecture that innate 'rules of thought', if they exist at all, are much simpler and 'deeper' in the sense of farther away from our actual reasoning in linguistic terms.) While constructing proofs in an unconscious way, we apply heuristic methods that depend on our intuitions, and the latter are the result of our past experiences within the relevant domain — in our case the actual proofs and reasonings we came across and our attempts to reproduce and imitate them. These intuitions will usually be governed by rules that are unconsciously applied and, more often than not, unconsciously acquainted. In other words, the heuristic methods we unconsciously apply in searching for proofs will be directly determined by (a specific part of) the culture in which we grew up, and indirectly by the structure of our languages and of our brains (think about the difference between humans and computer programs in accessing and organizing memory).

It would be a mistake to classify the problem of finding natural proof-heuristics as merely pedagogical or psychological. Some aspects of such methods will be contingent on physiological and cultural matters, but to study them involves a number of systematic problems. The fact that we are not able at present to draw a neat borderline between the two, should by no means prevent us from tackling the problem. The objection that we do not dispose of a systematically obtained set of empirical data should be replied to in a similar way. There is an impressive amount of competence from teaching logic courses. It seems quite sensible to start building hypotheses from this.

If a heuristic method embodies *insight*, it will first of all lead to a proof whenever there is an algorithm for finding one. Next, the method should be contextual: the way in which it reacts should depend on the actual problem, viz. the set of premises, the formula to be derived, the formulas already derived, intermediate goals introduced in view of the formula to be derived, etc. This is why heuristic methods deriving from standard tableau or resolution methods embody minimal insight only. Thirdly, given a specific set of inferential means (rules, axioms, ...) a heuristic method will in general lead to shorter proofs according as more insights are built into it. To illustrate all this, I list three Fitch-style proofs of $(t \vee p) \supset (r \& q)$, $s \supset {\sim}(r \vee u) \vdash s \supset {\sim}p$ according to increasing insight. (They are in terms of the rules enumerated in section 3, but that will not trouble any logician right now; in the fourth column, goals are recorded).

1	$(t \vee p) \supset (r \& q)$	PREM	
2	$s \supset {\sim}(r \vee u)$	PREM	$\triangle s \supset {\sim}p$
3	│ ${\sim}(s \supset {\sim}p)$	HYP	$\triangle$inconsistency
4	│ s	3; NI	
5	│ ${\sim}{\sim}p$	3; NI	
6	│ p	5; DN	
7	│ ${\sim}(r \vee u)$	2, 4; MP	
8	│ ${\sim}r$	7; ND	
9	│ ${\sim}u$	7; ND	
10	│		$\triangle t \vee p$
11	│ │ ${\sim}(t \vee p)$	HYP	$\triangle$inconsistency
12	│ │ ${\sim}t$	11; ND	
13	│ │ ${\sim}p$	11; ND	
14	│ │ p	6; REIT	
15	│ $t \vee p$	11, 13, 14; RAH	
16	│ $r \& q$	15, 1; MP	
17	│ r	16; ND	
18	$s \supset {\sim}p$	3, 8, 17; RAH	

1	$(t \vee p) \supset (r \& q)$	PREM	
2	$s \supset {\sim}(r \vee u)$	PREM	$\triangle s \supset {\sim}p$
3	│ s	HYP	$\triangle {\sim}p$
4	│ ${\sim}(r \vee u)$	3, 2; MP	
5	│ ${\sim}r$	4; ND	
6	│ ${\sim}u$	4; ND	
7	│ │ p	HYP	$\triangle$inconsistency
8	│ │		$\triangle t \vee p$
9	│ │ $t \vee p$	7; ADD	
10	│ │ $r \& q$	9, 1; MP	
11	│ │ r	10; SIM	
12	│ │ ${\sim}r$	5; REIT	
13	│ ${\sim}p$	7, 11, 12; RAH	
14	$s \supset {\sim}p$	3, 15; CP	

1	$(t \vee p) \supset (r \& q)$	PREM	
2	$s \supset \sim(r \vee u)$	PREM	$\vartriangle s \supset \sim p$
3	$p \supset (r \& q)$	1; DIE	
4	$p \supset r$	3; ICE	
5	$(r \vee u) \supset \sim s$	2; CPOS	
6	$r \supset \sim s$	5; DIE	
7	$p \supset \sim s$	4, 6; TRA	
8	$s \supset \sim p$	7; CPOS	

A few remarks are at hand. That a proof displays more insight is related to such things as the following: (i) it is shorter, (ii) it contains less superfluous steps — steps 8 and 11 of the first proof and step 6 of the second proof are superfluous — (iii) the introduction of hypotheses is postponed or eliminated, especially for hypotheses from which an inconsistency has to be found, (iv) it proceeds in a more goal-oriented way, and (v) it exemplifies more complex rules of inference. That the second proof proceeds in a more goal-directed way than the first is illustrated by step 9 (the goal is derived rather than arrived at by means of a subproof).

As the consecutive proofs are increasingly intelligent, it should be possible to extend the heuristic method resulting in the first proof into the one resulting in the second, and to extend the latter heuristic method into the one resulting in the third proof. By extending a heuristic method, one adds instructions that will postpone some (contextual) moves of the original method in favour of new types of moves (that are in general more efficient with respect to reaching the goal). Let me clarify at once why it is not desirable to introduce from the start a heuristic method that embodies maximal insight.

Apart from the requirements listed in the fourth paragraph of this section, the heuristic methods I have in mind should further people's insights in searching for proofs. By consciously applying such heuristic method, one's intuitive skills to unconsciously arrive at proofs should be improved. Again, this is something we have no systematically gathered data about. Again, I have to refer to the competence compiled by teaching logic courses. Nevertheless, few people will doubt in principle that such skills exist and may be improved, thus leading to more intelligent proofs. This is why it makes sense to devise consecutive heuristic methods as meant in the previous paragraph. The first method is meant for people with low insights. Once the latter have been sufficiently increased by applying the first method, one will *see* proof problems in a different way and perform a growing number of operations unconsciously. In view of this, one may start applying the second heuristic method, which now will look less complicated than it would have done originally. And so on for further extensions.

The fact that, in the present paper, I restrict my attention to classical (propositional) logic — henceforth PC — presents a special difficulty. In most every-day reasoning, we employ relevant implications rather than the material one. Even in contexts in which we do apply classical logic (mathematics, meta-theoretical proofs, etc.) we seem to favour inferences that are correct for relevant implications. (If you doubt this, just study a sufficient amount of examples). For this reason I shall, in the subsequent sections, whenever possible introduce rules of inference that are correct for relevant implications, thus pushing irrelevant properties of material implication into a minimal number of rules. Needless to say, this will have effects on the heuristic methods presented.

2. *Decisions on the format of proofs*

In order to illustrate what I have to say on heuristic methods, I need a formal system. I shall opt for the one that I gradually arrived at during twenty years of logic teaching. Obviously, this system may be replaced by another. The heuristic methods will be replaced accordingly, but the general principles will remain unchanged. Still, it seems appropriate to make some comments on the specific choice, especially as the choice itself has been influenced by thinking about and tinkering with heuristic methods.(2)

The system chosen is in Fitch style, with lines consisting of a line number and all or some of the following: a formula, a justification and a (general or local) goal. The dependence of the formula on premises and hypotheses is not recorded — compare with the Lemmon variant. In the usual finite case, all premises are listed at the top. The last line containing a premise and all lines containing a hypothesis need contain a goal. Analyzing a goal — see section 4 — may result in a line containing only a line number and a goal. *The goal* (at a stage) is the last goal that has not been attained (at that stage). The *depth of line* 1 is always(3) 0; the depth of other lines is

(2) See D. Batens, *Logicaboek. Praktijk en theorie van het redeneren*. Leuven/Apeldoorn, Garant, 1992.

(3) A formula occurring on line 1 is always a premise; we need a goal in order to introduce a hypothesis.

explicitly determined by the rules of the formal system. The *depth of a formula* and the *depth of a goal* are identical to the depth of the line on which they occur. The *depth of a subproof* is the depth of its first line (and of its hypothesis and goal). A *subproof* started by a hypothesis A with depth d is *closed* by the first subsequent formula with depth d − 1. The formulas of a subproof with depth d are those occurring with depth d between the hypothesis (included) and the formula closing it. A formula is *available* iff it occurs in the proof outside the closed subproofs. Uninterrupted vertical lines drawn at left in the second column mark subproofs (and show the depth of the lines of the proof).

This proof format is close to every-day reasoning in that, apart from the formulas in the proof, all that has to be remembered is the list of hypotheses of non-closed subproofs and the list of goals that have not yet been reached. The advantage of listing goals (other than the main one) is that, once a goal has been reached, it is obvious where we have to go next, viz. after the last goal not yet attained.

The only other (general) choices concern the rules of inference. In view of the aim to improve on people's inferential competence, a seizable set of derivable rules of inference is introduced, including all variants that people would only keep apart at the price of dull memorizing. (E.g., $A \supset \sim B$, B / $\sim A$ and other variants of Modus Tollens will be listed next to $A \supset B$, $\sim B$ / $\sim A$; Modus Tollens is thus turned in what it means for laymen anyway: from $A \supset B$ and the opposite of B, to derive the opposite of A).

3. *A formal system*

Given the above conventions and the obvious definitions of $A_1, \ldots, A_n \vdash B$ and $\vdash B$, the formal system is defined by the following rules; the depth of a line is the same as that of the previous line, except where explicitly stated.

Primitive rules:

PREM: A premise may be added to the proof as the first formula or after a formula with depth 0.[4]

HYP: An arbitrary formula may be added to the proof with a depth 1

[4] There is nothing wrong in allowing premises to be added within a subproof, but there is no need for doing so.

higher than that of the previous line.

REIT: Any available formula may be added to a subproof.

MP: $A \supset B$, A / B

CP: From a subproof with depth r that starts with the hypothesis A and ends with B, to derive $A \supset B$ with depth $r-1$.

SIM: A&B / A; A&B / B

CONJ: A, B / A&B

DIL: $A \vee B$, $A \supset C$, $B \supset C$ / C

ADD: A / $A \vee B$; B / $A \vee B$

EE: $A \equiv B$ / $A \supset B$; $A \equiv B$ / $B \supset A$

EI: $A \supset B$, $B \supset A$ / $A \equiv B$

DN: $\sim\sim A$ / A

RAA: $A \supset B$, $A \supset \sim B$ / $\sim A$

I at once list the *derived rules*:

TRA: $A \supset B$, $B \supset C$ / $A \supset C$

ICI: $A \supset B$, $A \supset C$ / $A \supset (B\&C)$

ICE: $A \supset (B\&C)$ / $A \supset B$; $A \supset (B\&C)$ / $A \supset C$

DII: $A \supset C$, $B \supset C$ / $(A \vee B) \supset C$

DIE: $(A \vee B) \supset C$ / $A \supset C$; $(A \vee B) \supset C$ / $B \supset C$

DIL: $A \vee B$, $A \supset C$ / $C \vee B$; $A \vee B$, $B \supset C$ / $A \vee C$ (variants)

DPAC: Arbitrarily many applications of permutation:

$\ldots \vee (A \vee B) \vee \ldots$ / $\ldots \vee (B \vee A) \vee \ldots$

association:

$\ldots \vee ((A \vee B) \vee C) \vee \ldots$ / $\ldots \vee (A \vee (B \vee C)) \vee \ldots$

(and vice versa) and contraction:

$\ldots \vee (A \vee A) \vee \ldots$ / $\ldots \vee A \vee \ldots$ (and vice versa).

DIST: $A\&(B \vee C)$ / $(A\&B) \vee (A\&C)$; $(A \vee B)\&C$ / $(A\&C) \vee (B\&C)$; $A \vee (B\&C)$ / $(A \vee B)\&(A \vee C)$; $(A\&B) \vee C$ / $(A \vee C)\&(B \vee C)$; the converses of all of these.

MT: $A \supset B$, $\sim B$ / $\sim A$; $\sim A \supset B$, $\sim B$ / A; $A \supset \sim B$, B / $\sim A$; $\sim A \supset \sim B$, B / A

CPOS: $A \supset B$ / $\sim B \supset \sim A$; $A \supset \sim B$ / $B \supset \sim A$; $\sim A \supset B$ / $\sim B \supset A$; $\sim A \supset \sim B$ / $B \supset A$

RAA: $\sim A \supset B$, $\sim A \supset \sim B$ / A (variant)

RAH: From a subproof with depth r that starts with the hypothesis A, respectively $\sim A$, and contains both B and $\sim B$, to derive $\sim A$, respectively A, with depth $r-1$.

DS: $A \vee B$, $\sim A$ / B; $A \vee B$, $\sim B$ / A; $\sim A \vee B$, A / B; $A \vee \sim B$, B / A
DN: A / $\sim\sim A$ (variant)
NC: $\sim(A \& B)$ / $\sim A \vee \sim B$
ND: $\sim(A \vee B)$ / $\sim A$; $\sim(A \vee B)$ / $\sim B$
NI: $\sim(A \supset B)$ / A; $\sim(A \supset B)$ / $\sim B$
NE: $\sim(A \equiv B)$ / $A \vee B$; $\sim(A \equiv B)$ / $\sim A \vee \sim B$

In order to reduce the number of reiterations, we allow rules to operate on all available formulas (and not only on formulas actually occurring in the current subproof); both CP and RAH may, however, require applications of REIT.

4. *Heuristic moves*

Given that the search for a proof of B from zero or more premises $A_1, \ldots, A_n$ is a goal-directed activity, there seem to be four kinds of steps that one may take within a proof: (i) to analyze available formulas in order to understand better what exactly is given, (ii) to analyze the goal in order to find out what exactly has to be proven, (iii) to derive a step that brings one closer to the goal, and (iv) to facilitate the search for the goal by introducing a hypothesis and its associated (local) goal. We shall soon see that one sometimes needs a further step, viz. (v) to introduce a goal in order to analyze an available formula.

$\sim\sim A$	A	DN
A&B	A, B	SIM
$A \equiv B$	$A \supset B$, $B \supset A$	EE
$(A \vee B) \supset C$	$A \supset C$, $B \supset C$	DIE
$A \supset (B \& C)$	$A \supset B$, $A \supset C$	ICE

TABLE I. Analysis of a formula

In order to analyze an available formula A, we need to derive from A one or more simpler formulas from which A itself may be derived. There are basically two cases. In the *first*, the formula itself is equivalent to a single simpler formula or to the conjunction of two simpler formulas. I list some examples in Table I in which the three columns contain respectively the formula to be analyzed (printed in bold face), the analyzing formulas, and the rule justifying the derivation of the formulas in column two from the formula in column one. In the *second case*, the formula is analyzed by combining it with other formulas. Some obvious examples are listed in Table II. Applications of dilemma are somewhat special in that the disjunction functions as an auxiliary formula to analyze two implications at once (but the disjunction itself does not get analyzed).

A ∨ B, ~A	B	DS
A ∨ B, ~B	A	DS
A ⊃ B, A	B	MP
A ⊃ B, ~B	~A	MT
A ⊃ C, **B ⊃ C**, A ∨ B	C	DIL

TABLE II. Analysis of a formula by means of an auxiliary formula

It is easily seen that all formulas printed in bold face in the first column of both tables are analyzed by those in the second in the sense that the former are themselves derivable from the latter. This means that we are facing a genuine analysis: that a formula is available reduces to one or more simpler formulas being available.

Analyzing a goal is quite similar: we introduce as a subgoal a formula that, alone or together with other formulas, will enable us to reach the goal. Here we should be careful to introduce only subgoals for which we have a warrant that they are derivable from the available formulas. Some examples are listed in Table III. The second row of that table should be read as: if A&B is the (last non-attained) goal and A is available, then introduce B as the goal. By consecutively applying, e.g., the instructions in the first two rows, we will be able to obtain A&B as soon as the two subgoals have been obtained.

I pause for a moment to remark that, with respect to the two moves considered up to now, SIM is only useful for analyzing premises, whereas CONJ is only useful for analyzing goals; ADD is useful for neither.

Δ**A&B** (A not available)	ΔA	(CONJ)
Δ**A&B**, A	ΔB	(CONJ)
Δ**A≡B** (A⊃B not available)	ΔA⊃B	(EI)
Δ**A≡B**, A⊃B	ΔB⊃A	(EI)

TABLE III. Analysis of the goal

Introducing hypotheses with respect to a goal is a simple matter. The only three cases are listed in Table IV. The first line of Table IV should be read as: where A⊃B is the goal, introduce A as a hypothesis and B as the associated goal. The procedure is fully transparent: as soon as the newly introduced goal is reached, we have to apply CP (for the first row) or RAH (for the two others) in order to obtain the goals listed in the first column.

Δ**A⊃B**	| A, ΔB	HYP
Δ~**A**	| A, Δinconsistency	HYP
Δ**A** (A not of the form B⊃C or ~B)	| ~A, Δinconsistency	HYP

TABLE IV. Starting a subproof in function of the goal

To derive a step in view of the last goal is actually the most complicated move because here all (non-structural) rules of inference may be applied — the steps themselves are rather simple though. We again should distinguish between two cases. In the *first* we *derive* the goal in one single step. I list a few dull examples in Table V. This move may be performed by anyone; all that is required is that one runs through the rules of inference and compares the available formulas and the goal. This kind of exercise makes one acquainted with the rules of the formal system (and hence, if the latter is well-devised, with the meaning of the logical terms).

Quite different and much more interesting is the *second case* in which we

▵A ∨ B, A	A ∨ B	ADD
▵A&B, A, B	A&B	CONJ
▵A, B ⊃ A, B	A	MP

TABLE V. Derive the goal from available formulas

search for a formula in view of the goal. Most of the competence for finding proofs derives from intuitive mechanisms that enable one to perform this move (in an unconscious way). Most applications of this move lead to searching for formulas for the derivability of which we have no warrant. Moreover, any goal might be obtained in different ways and from different formulas; e.g., A ∨ B might be obtained from A, from B, from C ∨ B and C ⊃ A, from C ⊃ (A ∨ B) and C, etc. To keep the number of searches under control, it seems wise first to distinguish between two subcases. Let A be the formula searched for. In the first subcase, the form of A suggests a set of formulas to be searched for; in the second subcase, we select the formulas searched for by comparing A with available formulas. More important, however, is that we find, for both subcases, a way to sensibly restrict the formulas to be searched for.

?A ∨ B	?A	ADD
?A ∨ B	?B	ADD
?A&B	?A, ?B	CONJ
?A, B ⊃ A	?B	MP
?A, B ∨ A	? ~ B	DS
?A ⊃ B, C ⊃ B	?A ⊃ C	TRA
?A ⊃ B, A ⊃ C	?C ⊃ B	TRA

TABLE VI. Search a formula in function of another

As one's inferential competence grows, one is able to larger iterations of the present move. For example, if the goal is A ∨ B, one may search for A;

next, if $C \supset A$ is available, one may search for C; etc. Because of the iterations, we have to include the steps we previously classified as analyzing the goal. Indeed, as we have no warrant that formulas searched for are derivable from the available formulas, we cannot list them as goals and hence they will not be analyzed by the steps listed in Table III. I enumerate some examples of searches in Table VI. Question marks indicate formulas searched for; some question marks in the first column may actually indicate goals. The fourth line should be read as: If one searches for A and $B \supset A$ is available, then search for B.

Finally, we come to the unexpected complication that we sometimes have to introduce a goal in order to analyze an available formula. Let us consider a simple example of such a proof.

1	$(p\&q) \supset \sim(q\&(r \supset r))$	PREM	$\triangle p \supset \sim q$
2	$\mid p$	HYP	$\triangle \sim q$
3	$\mid\mid q$	HYP	$\triangle$inconsistency

In view of all previously considered moves, we are locked (please check). There is no way to analyze the premise except by introducing a goal. The goal may be either p&q (in view of MP) or $q\&(r \supset r))$ (in view of MT). I present the proof obtained by proceeding according to the first option. The second option equally leads to the completion of the proof, as the reader may easily check.

4	$\mid\mid\mid$		$\triangle p\&q$
5	$\mid\mid\mid p\&q$	2, 3; CONJ	
6	$\mid\mid\mid \sim(q\&(r \supset r))$	5, 1; MP	
7	$\mid\mid\mid \sim q \vee \sim(r \supset r)$	6; NC	
8	$\mid\mid\mid \sim(r \supset r)$	7, 3; DS	
9	$\mid\mid\mid r$	8; NI	
10	$\mid\mid\mid \sim r$	8; NI	
11	$\mid \sim q$	3, 9, 10; RAH	
12	$p \supset \sim q$	2, 11; CP	

Given the rules of inference of the previous section, the only formulas to which it might be necessary to apply the present move are implications and disjunctions. All possible cases are listed in Table VII.

The difficulty with the present move is that we should take care that the newly introduced goal is derivable from the available formulas. Also, it

$A \supset B$	$\vartriangle A$	MP
$A \supset B$	$\vartriangle \sim B$	MT
$A \vee B$	$\vartriangle \sim A$	DS
$A \vee B$	$\vartriangle \sim B$	DS

TABLE VII. Introducing a goal in order to analyze a formula

turns out that it is not always sufficient to perform this move once. Sometimes we have to consecutively introduce a goal with respect to several formulas in order to get them analyzed. In articulating a heuristic method that contains this move, we shall face a double task: to warrant that the goal is attainable and to show that the method constitutes an algorithm for PC-derivability.

At present we are sufficiently equipped to move to the first heuristic method for PC-proofs.

5. *An algorithmic heuristic method*

I first present a heuristic method HM0 that is not natural. Its interest lies in the fact that the proof of its algorithmic character is easy and that subsequent heuristic methods are extensions of it; starting with HM0 greatly facilitates the meta-theory for all subsequent heuristic methods.

To simplify the description of HM0, I first define "analyzed formula" in an exact way. To this end I refer to Table VIII, in which !A denotes any 'opposite of A', viz. $\sim A$ but, where $A = \sim B$, also B; an available formula of a form mentioned in the left column is analyzed iff the corresponding formulas of the

$A \& B$	A and B
$A \vee B$	A or B
$A \supset B$	!A or B
$A \equiv B$	$A \supset B$ and $B \supset A$
$\sim\sim A$	A
$\sim(A \& B)$	$!A \vee !B$
$\sim(A \vee B)$	!A and !B
$\sim(A \supset B)$	A and !B
$\sim(A \equiv B)$	$A \vee B$ and $!A \vee !B$

TABLE VIII. Analyzed formulas

forms listed in the right column are available.

As for all heuristic methods considered, the problem will be to find a proof of B from premises A_1, ..., A_n ($n \geq 0$) — for obvious reasons, I only consider finite sets of premises. HM0 consists of an ordered set of instructions (referred to by their names for future use) that should be applied according to the following

CONVENTION ON ORDER

If an instruction leads to a change to the proof, one moves back to CHECK FOR GOAL; otherwise one moves to the next instruction.

Here is the list of instructions:

(1) START: Write down all A_i by application of PREM; write B as the goal on the line containing the last premise or on the first line if there are no premises. (This results in stage 1 of the proof).
(2) CHECK FOR GOAL: Check whether the goal (a formula or an inconsistent pair of formulas) has been reached. If the main goal of the proof is reached, stop; if another goal is reached, strike through the triangle in front of the goal (thus indicating that the previous goal has become 'the goal').
(3) DERIVE GOAL: Obtain the goal (possibly an inconsistency) by REIT whenever possible; else obtain it by RAH, whenever possible.
(4) ANALYZE FORMULA: Apply as much as possible SIM, EE, the DN-variant $\sim\sim A$ / A, NI, NC, ND, NE, MP, MT, DS, RAA, and DIL to non-analyzed available formulas.
(5) HYP FROM GOAL: If the goal is a formula A, add $\sim A$ as a hypothesis with an inconsistency as its associated goal.
(6) GOAL FROM FORMULA: If a formula of the form $A \supset B$ has not been analyzed and A is not a goal (that has not been reached), introduce A as the goal; if a formula of the form $A \vee B$ has not been analyzed and $\sim A$ is not a goal (that has not been reached), introduce $\sim A$ as the goal.

We need some further terminology in order to prove that HM0 is an algorithm for finding PC-proofs. By a *final stage* of a proof constructed by HM0 I shall mean either a stage at which the desired proof is obtained or a stage that is left unchanged by HM0. I write "a final stage" because HM0 is not strictly deterministic (see ANALYZE FORMULA and GOAL FROM FORMULA). Let G_i be the goal at stage i. (I recall that G_i, the last goal that has not been reached, is not necessarily introduced at the last line of stage i.) Let the *characteristic set* of stage i be denoted by Ω_i and consist of all formulas (i) available at that stage and (ii) the depth of which is not larger than the depth of G_i. (If, at stage i, the main goal of a subproof with depth d is available and CHECK FOR GOAL has been executed, the previous goal becomes the goal; as its depth is always $d-1$, the formulas of the subproof do not belong to Ω_i.) The *characteristic statement* of stage i is "$\Omega_i \vdash G_i$". If the goal is an inconsistency, I shall write "$\Omega_i \vdash$ inconsistency". In the present section, I only consider proofs arrived at by HM0.

Lemma 1. If $A_1, \ldots, A_n \vdash B$, then the characteristic statement of any stage of the proof B from $A_1, \ldots, A_n$ holds true.

Proof. Supposing that $A_1, \ldots, A_n \vdash B$, we proceed by induction on the stage i of the proof (the stage reached by executing an instruction). We have to consider six cases corresponding to the instructions of HM0.

(i) After executing START,

$$\Omega_1 \vdash G_1$$

is nothing but

$$\{A_1, \ldots, A_n\} \vdash B$$

(ii) If CHECK FOR GOAL has been executed, we have to consider several subcases. The characteristic statement is not modified (nor is anything else, for that matter) if the *main* goal of the proof is reached. So, let us consider other goals being reached.

(ii/i) If the goal reached, viz. G_{i-1}, has the same depth as the previous goal G_j (the goal of the stage j just before goal G_{i-1} was introduced), then we have:

$$\Omega_j \vdash G_j$$

$$\Omega_{i-1}\ (= \Omega_j \cup \{\ldots, G_{i-1}\}) \vdash G_{i-1} \qquad (i-1 > j)$$

$$\Omega_i\ (= \Omega_j \cup \{\ldots, G_{i-1}\}) \vdash G_i\ (= G_j)$$

Clearly, the characteristic statement of stage i is justified by the characteristic statement of stage j. Remark that G_{i-1} always is a formula[5] and that it does not make any difference whether G_{i-1} was reached by applying REIT, RAH, or another rule.

(ii/ii) If the goal reached, viz. G_{i-1}, is the main goal of a subproof — hence G_{i-1} always is "inconsistency" — consider the stage j immediately preceding the start of the subproof. Obviously the depth of G_j is 1 lower than the depth

[5] An inconsistency is only introduced as the goal associated with a hypothesis and hence its depth is always 1 higher than the depth of the previous goal.

of G_{i-1}, and we have:

$\Omega_j \vdash G_j$

Ω_{i-1} $(= \Omega_j \cup \Phi) \vdash G_{i-1}$ $\quad (i-1 > j)$

Ω_i $(= \Omega_j) \vdash G_i$ $(= G_j)$

Φ contains the formulas that belong to the subproof. The characteristic statement of stage i is justified by the characteristic statement of stage j. Again, it does not make any difference whether G_{i-1} was reached by applying REIT, RAH, or another rule.

(iii) After executing DERIVE GOAL,

$\Omega_{i-1} \vdash G_{i-1}$

warrants

Ω_i $(= \Omega_{i-1} \cup \{C\}) \vdash G_i$ $(= G_{i-1})$

(if the goal was derived by REIT, $C \in \Omega_{i-1}$; if the goal is derived by RAH, $C = G_{i-1}$).

(iv) After executing ANALYZE FORMULA,

$\Omega_{i-1} \vdash G_{i-1}$

warrants

Ω_i $(= \Omega_{i-1} \cup \{C\}) \vdash G_i$ $(= G_{i-1})$

(v) After executing HYP FROM GOAL,

$\Omega_{i-1} \vdash G_{i-1}$

warrants

Ω_i $(= \Omega_{i-1} \cup \{\sim G_{i-1}\}) \vdash$ inconsistency

(vi) GOAL FROM FORMULA can only be executed (check HM0) if the goal is an inconsistency. (Otherwise, HYP FROM GOAL would be executed.) Hence,

$\Omega_{i-1} \vdash$ inconsistency

warrants

Ω_i $(= \Omega_{i-1}) \vdash G_i$. ■

Let the *complexity of a formula* A, C(A), be defined by: (i) where A is a propositional letter, $C(A) = 1$, (ii) $C(\sim A) = C(A)+1$, (iii) $C(A \supset B) = C(A \vee B) = C(A)+C(B)+2$, (iv) $C(A\&B) = C(A)+C(B)+4$, and (v) $C(A \equiv B) = 2 \times (C(A)+C(B)+4)$. Where Γ is the set of all atoms (propositional letters or the negation of such) in Ω_i and where $\Delta_A = \{B \mid B \in \Omega_i$ and $C(B) < C(A)\}$, we shall say that A is *distributed* at stage i iff $\Delta_A \cup \Gamma \vdash A$, and that A is *fully distributed* iff $\Gamma \vdash A$. (All atoms and all analyzed formulas are distributed; if $\sim p$ is available, $(p\&q) \supset r$ is fully distributed but not necessarily analyzed.)

Lemma 2. If an inconsistency is derivable from Ω_i and all members of Ω_i are fully distributed, then Ω_i contains an explicit inconsistency.
Proof. Suppose that the antecedent is true. Let Γ be the set of atoms in Ω_i and let p_j be a propositional letter that does not occur in Γ. All members of Ω_i are derivable from Γ. Hence $\Gamma \vdash p_j \& \sim p_j$. If Γ does not contain an explicit inconsistency, then it is possible to substitute $p_j \vee \sim p_j$ for any propositional letter A such that $A \in \Gamma$ and to substitute $p_j \& \sim p_j$ for any propositional letter A such that $\sim A \in \Gamma$. But then $\vdash p_j \& \sim p_j$, which is impossible. ■

Lemma 3. Applying HM0 to construct a proof of B from $A_1, \ldots, A_n$ either leads to such proof or else results in a final stage i such that all members of Ω_i are fully distributed and G_i is an inconsistency.
Proof. Suppose that, applying HM0, *we do not find the required proof.* In view of HYP FROM GOAL we are bound to reach a stage i at which $\sim B$ is added to the proof by HYP with an inconsistency as its associated goal. Let the depth of this subproof be d_i. If we continue to apply HM0, the instruction ANALYZE FORMULA warrants that all available formulas of the forms $C \& D$, $C \equiv D$, $\sim\sim C$, $\sim(C \supset D)$, $\sim(C \& D)$, $\sim(C \vee D)$, and $\sim(C \equiv D)$ are distributed, in other words, all non-atomic formulas except (possibly) for formulas of the form $C \supset D$ and $C \vee D$.

Consider some stage j ($\geq$ i) at which some available formulas are not fully distributed, all non-distributed formulas are of the form $C \supset D$ or $C \vee D$, the goal is an inconsistency, and the depth of this goal is d_j; clearly $d_j \geq d_i$. For some formula $E \supset F$ (respectively $E \vee F$) that is not fully distributed, the instructions GOAL FROM FORMULA and HYP FROM GOAL will force us to introduce the goal E (respectively $\sim E$) and next to introduce the hypothesis $\sim E$ (respectively $\sim\sim E$) with depth $d_j + 1$ and with an inconsistency as the associated goal; $E \supset F$ (respectively $E \vee F$) is thereby distributed (at depth $d_j + 1$). If an inconsistency is later attained from the hypothesis, E (respectively $\sim E$) will be derived (at depth d_j) from the subproof, the goal will be replaced by G_j, viz. "inconsistency", and F will be derived from E and $E \supset F$ (respectively $\sim E$ and $E \vee F$); in this case $E \supset F$ (respectively $E \vee F$) is distributed (at depth d_j). Summarizing: $E \supset F$ (respectively $E \vee F$) will be distributed at depth d_j or at depth $d_j + 1$. By continuing to apply HM0 up to HYP FROM GOAL (which itself cannot be applied as the goal is an inconsistency), we reach a stage k at which the goal is an inconsistency and all non-distributed formulas, if any, are of the form $C \supset D$ or $C \vee D$.

By repeating the procedure described in the last paragraph, we shall ultimately reach a stage at which all available formulas are distributed and hence fully distributed. Indeed, whether the subproof is closed or not, the analysis of a formula A will at most require a finite number of steps and will, in the worst case, lead to adding one or two formulas. As the sum of the complexity of each of these formulas is smaller than the complexity of A, all formulas will ultimately be fully distributed. Once they are and an inconsistency is not obtained, a final stage is reached. ■

For a good understanding of the previous proof, it is necessary to realize that, if GOAL FROM FORMULA and HYP FROM GOAL lead to a subproof, some available formulas may be distributed by formulas in the subproof but become non-distributed when it is closed. Still, we are bound to reach a final stage at which all formulas are fully distributed because (i) if after executing GOAL FROM FORMULA with respect to A, a subproof with depth d was started by executing HYP FROM GOAL and next was closed, the subsequent application of MP or DS will result in A being distributed at depth $d-1$, and (ii) if a formula is distributed while the depth of the goal is d, it will remain distributed at least until the depth of the goal is $d-1$. (It is easily seen that this results in a tree that displays the finite fork property and the finite branch property, and hence is finite by Koenig's lemma.)

Theorem 1. HM0 will lead to a proof of B from $A_1, \ldots, A_n$ if there is one. Proof. Suppose that $A_1, \ldots, A_n \vdash B$, that HM0 is applied to obtain a proof of B from $A_1, \ldots, A_n$, and that the desired proof is not obtained. It follows from lemma 3 that HM0 leads to a final stage i at which all available formulas are fully distributed and at which the goal is an inconsistency. Then the characteristic statement, viz. "$\Omega_i \vdash$ inconsistency", holds true in view of lemma 1. As all available formulas at stage i are fully distributed, Ω_i contains an explicit inconsistency in view of lemma 2. But then the goal is obtained and hence i is not a final stage. ■

HM0 always results in a final stage; hence:

Corollary 1. HM0 is an algorithm for deciding whether there is a proof of B from $A_1, \ldots, A_n$.

6. *A natural heuristic method*([6])

The frequent use of the instruction HYP FROM GOAL, as described in the previous section, leads to particularly bad proof habits. Try it on freshmen; they will find proofs but not gain much insight from them. This is why, as a first natural heuristic method, one should introduce something like HM1, which is governed by the same CONVENTION ON ORDER as HM0 and consists of the following instructions:

(1) START
(2) CHECK FOR GOAL
(3) DERIVE GOAL (modified): Whenever possible, obtain the goal (possibly an inconsistency) by REIT; else by RAH or CP; else by a single application of another rule of inference. Pay attention especially to the DN-variant A / $\sim\sim$A, CONJ, ADD, EI, DPAC, DIST, TRA, and CPOS (the 'constructive' rules of inference).
(4) ANALYZE GOAL: Analyze the goal in function of CONJ and EI (see Table III).
(5) ANALYZE FORMULA
(6) HYP FROM GOAL (modified): Introduce a hypothesis and the associated goal according to Table IV.
(7) GOAL FROM FORMULA

As HM1 is an extension of HM0, it is an easy task to adapt the proofs of the previous lemmas and theorem (with HM1 replacing HM0). The proofs of lemma 2 and theorem 1 remain unchanged. The proof of lemma 1 has to be adapted as follows. After the execution of ANALYZE GOAL,

$\Omega_{i-1} \vdash G_{i-1}$

warrants

$\Omega_i \ (= \Omega_{i-1}) \vdash G_i$

because either G_{i-1} is C&D, in which case G_i is C or D, or G_{i-1} is $C \equiv D$, in which case G_i is $C \supset D$ or $D \supset C$. Also, the addition of ANALYZE GOAL does not change the situation arrived at after executing CHECK FOR GOAL (see subcase (i/i) of the proof of lemma 1). Finally, the execution of the modified instruction HYP FROM GOAL leads to a situation in which either

([6]) The natural heuristic methods discussed in this paper (and extensions of them) form the basis of a computer programme (and where arrived at while developing it) — see D. Batens, *Logicaboek. Computerprogramma's*. Leuven/Apeldoorn, Garant, 1992.

$\Omega_{i-1} \vdash G_{i-1}$

warrants

$\Omega_i \ (= \Omega_{i-1} \cup \{!G_{i-1}\}) \vdash$ inconsistency

or

$\Omega_{i-1} \vdash G_{i-1} \ (= C \supset D)$

warrants

$\Omega_i \ (= \Omega_{i-1} \cup \{C\}) \vdash G_i \ (= D)$

The only change required in the proof of lemma 3 is connected to the modification to HYP FROM GOAL. First of all, if $B = C_1 \supset D_1$, HYP FROM GOAL will not lead to the hypothesis $\sim B$ with an inconsistency as the associated goal, but to the hypothesis C_1 with the associated goal D_1. If D_1 is not reached, however, the hypothesis $\sim D_1$ will be introduced with an inconsistency as the associated goal. Moreover, $C_1, \sim D_1 \vdash \sim(C_1 \supset D_1) = \sim B$. In other words, if there is a later stage at which both C_1 and $\sim D_1$ are fully distributed then $\sim B$ would have been fully distributed at that stage if it had occurred in the proof. Exactly the same reasoning applies to the case where executing GOAL FROM FORMULA with respect to some formula F leads to adding the goal E and $E = C_1 \supset D_1$; in the worst case, executing HYP FROM GOAL twice will lead to a stage at which F is distributed (in view of the hypotheses C_1 and $\sim D_1$) and the goal is an inconsistency. As a consequence, we shall reach a final stage at which all available formulas are fully distributed and the goal is an inconsistency (unless we find the proof).

I take HM1 to be the basic natural heuristic method for the Fitch-style system of 3. It is algorithmic, leads to natural proofs if it is applied to problems that are not too complex, and helps one to become acquainted with most of the inferential means provided by the (primitive and derived) rules. In contradistinction to HM0, the heuristic method HM1 attracts attention to (local) goals; it nevertheless relies sufficiently on the analysis of available formulas to reduce inferential problems to reasonably simple tasks.

7. *Improving on the analysis*

As I remarked in section 1, more intelligent proofs are usually shorter, they exemplify more complex rules, and the introduction of hypotheses is postponed or eliminated in them. This suggests at once that we may improve upon the quality of HM1 by modifying ANALYZE GOAL in such a way that the analysis also takes account of ICI and DII and by modifying ANALYZE FORMULA in such a way that it includes moves justified by ICE and DIE.

This will be the first set of modifications to HM1.

The second set of modifications concerns rules of inference that as yet do not play any role in the analysis of the goal or of available formulas: CPOS, DIST and DPAC. If it turns out to be impossible to derive or analyze the goal, we shall first recur to

> REFORMULATE GOAL: try to replace the goal A by a goal B from which A is derivable by CPOS, DIST or DPAC, provided that either B is available or derivable (in one step) from available formulas, or B may be analyzed according to ANALYZE GOAL.

Here are some illustrations. The instruction ANALYZE GOAL will not have any effect if the goal is $C \supset \sim(A \vee B)$, but it will if we first transform the latter to $(A \vee B) \supset \sim C$. Similarly for the goal $B \vee (A \& C)$ and its DIST-transform $(B \vee A) \& (B \vee C)$. Contextual example: $B \vee (A \vee C)$ is not derivable in one step from A, but its DPAC-transform $A \vee (B \vee C)$ is.

Similarly, if the analysis of some available formula fails, we shall first (before introducing a hypothesis) recur to

> REFORMULATE FORMULA: try to replace an available formula A by a formula B that is derivable from A by CPOS, DIST or DPAC, and that may be analyzed according to ANALYZE FORMULA.

If no available formula enables us to analyze $A \supset \sim(B \vee C)$, we shall apply CPOS, thus obtaining $(B \vee C) \supset \sim A$, which may be analyzed by DIE; $A \vee (B \& C)$ will be transformed by DIST to $(A \vee B) \& (A \vee C)$ for similar reasons. Contextual example: given $\sim A$, $(B \vee A) \vee C$ will be transformed to $A \vee (B \vee C)$.

The modifications discussed in the present section lead to a heuristic method HM2 which is governed by the CONVENTION ON ORDER and consists of the following sequence of instructions: (1) START, (2) CHECK FOR GOAL, (3) DERIVE GOAL, (4) ANALYZE GOAL (now taking account of ICI and DII), (5) REFORMULATE GOAL, (6) ANALYZE FORMULA (now taking account of ICE and DIE), (7) REFORMULATE FORMULA, (8) HYP FROM GOAL, and (9) GOAL FROM FORMULA. It is easily seen that, in general, the application of HM2 leads to proofs that are more intelligent in the above sense.

In order to adapt the lemmas and theorem (from HM1 to its extension HM2), no change is needed to the proof of lemma 1 and theorem 1 whereas the required modifications to the proofs of lemmas 2 and 3 are strictly straightforward.

8. *Proceeding in a more goal-directed way*

Proofs that are more goal-directed, and hence avoid superfluous steps, are arrived at by a further modification. The idea is that we *search* for formulas, even if we are not sure that they are actually derivable. The search will be *iterative*: if the formula searched for is not available, we shall search for a formula from which it may be obtained.

A search may be *contextual* or non-contextual. In case we want to obtain $A \vee B$, to search for A is an example of a non-contextual search whereas to search for C in the presence of $C \supset (A \vee B)$ is an example of a contextual search. The three first rows of Table VI concern non-contextual searches, the four others concern contextual searches. As one might expect, it proves more efficient to search contextually first.

Most rules of inference induce several search moves. Drawing up the (long and dull) table is left to the reader. It is worth explaining, however, that a contextual search is set up as follows: the searched formula is compared to each available formula *in view of the form of the latter*. Suppose that we focus on an available formula $A \supset B$. If we search for $\sim B \supset \sim A$, we apply CPOS; if we search for B, we engage in a search for A; if we search for $A \supset C$, we engage in a search for $C \supset B$; etc. If the contextual search fails with respect to all available formulas, we start a non-contextual search in view of the form of the formula searched for.

The problem with iterative searches is that they might go on forever. For this reason, it is necessary to introduce some restrictions on starting new searches. The first is that we always search for formulas that are *not more complex*(7) than the formula searched for and the available formula in focus. Searching for A and focusing on B, we do *not* engage in a search for $B \supset A$. This restriction first prevents us from arriving at proofs that are superfluously complex. If the restriction were not introduced, a search for A in the presence of the available formula B would lead us to also search for $B \supset A$, $\sim A \supset \sim B$, $\sim B \vee A$, ...; the search for $B \supset A$ would then lead us to search for $B \supset (B \supset A)$, $\sim(A \supset B) \supset \sim B$, $\sim B \vee (A \supset B)$, ..., and, if C

(7) An exception may be allowed for TRA; if we search for $p \supset (q \vee r)$ and $p \supset (t \vee u)$ is available, it is sensible to search for $(t \vee u) \supset (q \vee r)$. There are no other exceptions and the present one is not important enough to be circumvented by modifying the general rule.

is available, also for $C \supset (B \supset A)$, $\sim(A \supset B) \supset \sim C$, $\sim C \vee (A \supset B)$, etc. If such searches lead to the desired result, viz. A, there usually is a simpler way to derive A as well. This first restriction moreover prevents us from continuously stumbling upon circular searches and searches for inconsistencies (that are unlikely to succeed). A search for A in the presence of the available formula B would make us search for $\sim B \vee A$, and this would make us search for A again, and also for $\sim B$.

The second restriction is that a new search is started *in view of a single rule of inference*, not in view of a combination of such rules. To take a simple example: if we search for A and focus on the available formula $B\&(A \vee C)$, there is no single rule of inference that enables us to engage in a new search; only after we *analyzed* this formula into B and $A \vee C$, the latter will make us search for $\sim C$. (The rationale for the present restriction is to avoid complications; if one is able to handle them, there is nothing wrong with abolishing the restriction.) Given both restrictions, iterative searches lead to rather small search trees for most *but not all* examples of proofs.

The third restriction is that search paths should be prevented from becoming circular. Here is a simple example: if we search for A in the presence of both $A \vee B$ and $\sim A \vee \sim B$, we first will search for $\sim B$, and this will make us search for A again. The latter search path has to be broken off immediately. (We shall look only for other ways to arrive at $\sim B$.)

The three restrictions discussed up to now are *systematic* in nature and are justified in view of the efficiency of the heuristic method. Apart from them, there is a restriction that is related to a brute fact: the *complexity* of search trees someone is capable of handling. This depends *not only* on the available amount of short-time memory one is able to control. At each node of the search tree, there may be a large number of branches, and one has to remember the branches already tried out for each node on the present path. For contextual searches, the branch is characterized by the available formula in focus and by the move taken in view of the form of this formula. Let me give an example to illustrate the difficulty. Suppose we are searching for A and focus on the available formula $B \supset C$. We have to check for each of the following: is A derivable from $B \supset C$ by CPOS, DIE, or ICE, or should we search for another formula in view of MP ($A = C$), MT ($A = \sim B$), TRA ($A = B \supset D$ or $A = D \supset C$), ICI ($A = B \supset (C\&D)$ or $A = B \supset (D\&C)$), ... After each of these failed with respect to each available formula, we still have to engage in a non-contextual search for A (which, needless to say, may engage us in a contextual search at subsequent nodes).

There is no difficulty in handling a maximal complexity limit: once it is reached, one checks whether the formula searched for is derivable from an available formula but does not engage into a new search. Also, it is not difficult to see how one may train oneself in handling more complex search trees: practice is the means. The difficulty related to the complexity of search trees is that this complexity is hard to measure (as it has several dimensions) and that it is hard to compare the efficiency of increasing this complexity with the efficiency of a deeper analysis of the goal and the available formulas.

Given the above comments, we are in a position to articulate the first instruction concerning searches:

> SEARCH FOR GOAL: If the goal is a formula, start an iterative search for it according to the procedure determined by the three systematic restrictions and taking account of the maximal complexity of the search tree; if the goal is an inconsistency, start similar searches for the negation of available formulas.

There is a second reason why one might start a search tree, as appears from the following (first stage of a) proof:

1	$(p\&q) \supset ((s \supset t) \vee r)$	PREM	
2	p	PREM	
3	q	PREM	
4	$\sim r$	PREM	$\triangle s \supset t$

If we proceed according to HM2, we shall first introduce the hypothesis s, next the hypothesis $\sim t$, and next the goal p&q. However, it is obvious that a much more intelligent proof is found by consecutively applying CONJ, MP and DS. So, a rather simple aspect of the intelligent character of proofs is missing from HM2. This is repaired by searching for formulas that enable one to analyze available formulas (before new hypotheses are added). Here is the instruction:

> SEARCH FOR ANALYZING FORMULA: start an iterative search (as defined in SEARCH FOR GOAL) for formulas that enable one to analyze available and non-analyzed formulas.

The modifications discussed in the present section lead to a heuristic method HM3 which is governed by the CONVENTION ON ORDER and consists of the following sequence of instructions: (1) START, (2) CHECK FOR GOAL, (3) DERIVE GOAL, (4) SEARCH FOR GOAL, (5) ANALYZE GOAL, (6) REFORMULATE GOAL, (7) ANALYZE FORMULA, (8) REFORMULATE FORMULA, (9) SEARCH FOR ANALYZING FORMULA, (10) HYP FROM GOAL, and (11) GOAL FROM FORMULA.

It is typical for the two search instructions that, similar to ANALYZE GOAL, they enable us to arrive at sensible applications of rules of inference that take no part in the analysis of formulas; examples are ADD, CONJ, EI, DII, ...

Incidentally, there is no reason why the two changes discussed in the present section should be introduced together; SEARCH FOR GOAL might even be introduced before the instructions introduced in the preceding section. The main point I wanted to make is that, once one has gained the insights deriving from applying HM1, one will arrive at more intelligent proofs by moving closer to HM3.

The way in which the lemmas and theorems may be adapted to HM3 is obvious as searches only lead to adding formulas when they succeed.

9. *Further extensions*

It is straightforward that HM3 may still be extended in several ways. One of them is to combine several rules of inference, especially with respect to contextual searches, as in the case where one sees at once that DPAC and DS together enable one to derive A from $B \vee (A \vee C)$ and $\sim(B \vee C)$. It seems to me, however, that this does not constitute a good reason for studying further extensions of HM3. After all, I did all I promised to do, and there is no optimal or most-embracing heuristic method anyway.

A different question is whether some important aspect is still missing from HM3. The reader may feel there is, as it is not clear that superfluous (analyzing) moves have been prevented. As a first reply, let me point out that many such moves are prevented by HM3. This is related to the fact that, in many proofs, most of the analysis of available formulas will not result from executing ANALYZE FORMULA, but from executing (the earlier instruction) SEARCH FOR GOAL.

Still, it is correct that the efficiency in avoiding superfluous analyzing steps may be increased. In analyzing available formulas, we may proceed in a goal-directed way: we may first derive formulas that are *likely to be useful* with respect to searched formulas. If we search for p, then $q \supset p$ is likely to be useful, whereas $p \supset q$ is not. The reason for this is that, if q later becomes available, we will be able to derive p from q and $q \supset p$. At this point, the reader might object that, if $(p \supset q) \supset p$ later becomes available, we will be able to derive p from $p \supset q$ and $(p \supset q) \supset p$. This, however, is not a sensible way to proceed, because it will not enable us to classify some steps as more appropriate than others; any B is relevant to A, in that later $B \supset A$ may become available.(8) In the case of $q \supset p$, the matter is different because p will be derivable from it if the formula q, which is *less complex* than $q \supset p$, becomes available.

My suggestion then is that a formula B is likely to be useful with respect to a searched formula A, if we see a way to derive A from B with the help of a formula C that is not more complex than A or B. (One should not require that C be simpler than either A or B; clearly, $p \supset q$ is likely to be useful for deriving $p \supset r$ because $q \supset r$ would enable us to do so.) Of course, more accurate criteria are possible, e.g., in terms of positive and negative parts, but these are rather technical and farther away from natural heuristic methods. For this reason, I contend that the above suggestion will do. Formulated in a more precise way, my suggestion is that, in executing ANALYZE FORMULA, REFORMULATE FORMULA, and SEARCH FOR ANALYZING FORMULA, we first derive the formula that is most likely to be useful with respect to a searched formula. By proceeding in this way, we drastically *change the order* in which the instructions are executed: we first decide which analyzing step is most likely to be useful, and only thereafter try to arrive at it by the three instructions (in their order). In the remote case that no formula that is likely to be useful would be arrived at, one analyzes formulas irrespective of this criterion. But even then a proviso is at hand: there is no need to derive formulas which one sees to have no use. So, eventually, it may be sensible to introduce a hypothesis even if some available formulas are not analyzed or even distributed.

The upshot is that the instructions are executed according to the CONVENTION ON ORDER, but only in so far as the results of their execution are useful. However, one should only execute GOAL FROM FORMULA if the goal is an inconsistency — see the proof of lemma 3. No such proviso applies to SEARCH FOR ANALYZING FORMULA, which will take over most of the role

(8) The situation is even worse: in PC, p is derivable from $(p \supset q) \supset p$ alone. However, the objection in the text holds for other implications as well.

from GOAL FROM FORMULA for anyone who follows the rich heuristic methods discussed in the present section.

Department of Philosophy
Universiteit Gent
Rozier 44
9000 Gent (Belgium)