ARTIFICIAL INTELLIGENCE LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

M.I.T. AI Memo 857 **August 1985**

VARIABLE PRECISION LOGIC

by

Ryszard S. Michalski and Patrick H. Winston

Abstract. Variable precision logic is concerned with problems of reasoning with incomplete information and under time constraints. It offers mechanisms for handling trade-offs between the precision of inferences and the computational efficiency of deriving them. Of the two aspects of precision, the *specificity* of conclusions and the *certainty* of belief in them, we address here primarily the latter, and employ *censored production rules* as an underlying representational and computational mechanism. Such rules are created by augmenting ordinary production rules with an *exception* condition, and are written in the form *if A then B unless C*, where C is the exception condition.

From a *control* viewpoint, censored production rules are intended for situations in which the implication $A \Rightarrow B$ holds frequently and the assertion C holds rarely. Systems using censored production rules are free to ignore the exception conditions, when time is at premium. Given more time, the exception conditions are examined, lending credibility to initial, high-speed answers, or changing them. Such logical systems therefore exhibit variable certainty of conlusions, reflecting variable investment of computational resources in conducting reasoning. From a *logical viewpoint*, the *unless* operator between B and C acts as the exclusive-or operator. From an *expository viewpoint*, the *if A then B* part of the censored production rule expresses an important information (e.g., a causal relationship), while the *unless C* part acts only as a switch that changes the polarity of B to $\neg B$ when C holds.

Expositive properties are captured quantitatively by augmenting censored rules with two parameters that indicate the certainty of the implication *if A then B*. Parameter δ is the certainty when the truth value of C is unknown, and γ is the certainty when C is known to be false.

Acknowledgement. This research was done in part at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology and in part at the Department of Computer Science at the University of Illinois. Support for MIT's artificial-intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-80-C-0505. Professor Michalski's research at the University of Illinois is supported in part by the National Science Foundation under grant NSF DCR 84-06801 and in part by the Office of Naval Research under grant N00014-82-K-0186.

Variable Precision Logic Is Concerned with Specificity and Certainty

You cannot tell an ordinary logic-based reasoning system much about how you want it to do its job. You cannot give the following instructions, for example:

- Give me a reasonable answer immediately, even if somewhat general; if there is enough time, give me a more specific answer.
- Give me a reasonable answer immediately; if there is enough time, tell me you are more confident in the answer or change your mind, giving me another, better answer.

Suppose, for example, that you want to know what John is doing, given that it is Sunday. A quick answer is that he is *probably* working in his yard. A more *specific* answer, obtained by taking into consideration the time of year, is that he is *specifically* raking leaves. A more *certain* answer, obtained by noting nice weather, is that he is *certainly* working in his yard, rather than reading.

A system that gives more specific answers, given more time, is what we call a *variable-specificity system*. A system that gives more certain answers, given more time, is what we call a *variable-certainty system*. There can be various combinations of the two systems, reflecting the fact that specificity and certainty are inversely related: we can gain specificity at the expense of certainty, or we can gain certainty by sacrificing specificity. This point can be illustrated by going back to our example about what John is doing on Sunday. In this exammple, the less specific statement, "John is working outdoors," is more certain than the more specific statement, "John is working in his yard."

Variable specificity and variable certainty are two aspects of what we call *variable precision*. Thus, in general, we say that a variable-precision system is a system that exhibits either variable specificity or variable certainty or some tradeoff between the two.

The purpose of this paper is to show how we have extended ordinary logic so as to enable logic-based systems to exhibit variable precision in which certainty varies while specificity stays constant. As a vehicle to implement such a logic system we employ *censored production rules*, which are production rules with exceptions.

Before we explain censored production rules in detail, however, we discuss the need for rule-repair mechanisms, which is our second motive for extending logic with censors.

Censors Make Exceptions Explicit, Facilitating Rule Repair

The rules that we postuate for the world are normally tentative because there are few regularities that hold universally for any entity or class of entities. There is always a possibility that rules may have to be revised in the face of new facts or new happenings. Heraclitus's *panta rei*, i.e., all is in motion, applies not only to the outside reality but also to our mental representations of that reality. Because our knowledge is fluid and subject to modifications, a representation of that knowledge should make modifications easy and natural. A corolary is that changes to formal descriptions should closely reflect changes in our own thinking.

What then do we do when a rule (or a theory) which has worked well in the past does not work in some newly observed situation? There are several possibilities:

1. To consider the rule invalid, and ignore it in the future.
2. To continue to use the rule without change, realizing that using it will result in error occasionally.
3. To modifiy the rule, so that the rule applys correctly to all encountered situations.
4. To develop a new rule, substituting the new rule for the old one.
5. To remember the situations for which the rule does not work, treating them as exceptions.

All of these choices force a trade-off between estimated cost and estimated benefit. In some situations we may not be able to afford the time or other resources to make modifications to the rule before we need to use it. Also, which type of rule repair is best depends on the type of contradiction found to the rule.

Action 1, to invalidate the rule, is simple and prevents us from making errors, but may leave us handicapped. If the rule worked well in many cases, then invalidating it deprives us of the benefit of using it when it does work.

Action 2, to use the rule without change, is also simple. It preserves the benefit of employing it when it does work, but using it will lead to some errors which may be costly.

Action 3 calls for creating a new rule by modifying the old one, and action 4 calls for developing a new rule from scratch. If the modification to be made to a rule is small, then action 3 is the better choice. But if this modification is complicated or unclear, then action 4 is the better choice. In general, both actions lead to a better and more precise rule, but both require time and effort. In science, where standards for precision and certainty are high, one of these two actions is a usual choice, no matter the cost. The problem of incrementally refining rules to accomodate new facts is explored in Reinke and Michalski [1985].

If the exceptions are few and easy to remember, then action 5, to remember exception conditions, is a good choice. It preserves the usefulness of the old rule, but prevents us from making mistakes in situations recognized as exceptions. Even when there are more than a few exceptions, remembering them still may be the best action to take, particularly when it is not clear how to make changes to the old rule or how to create a new one.

Another situation when remembering exceptions may be the best choice is when a modified or completely new rule is significantly more complicated then the original rule. A simple rule with exceptions may be better then a complicated one without exceptions, particularly when the exceptions occur only rarely. When the number of exceptions grows, they may be generalized via the introduction of a new rule, reducing the overall complexity of the original rule.

From this point of view, the purpose of this paper is to introduce ideas centered on the exception-remembering approach to knowledge modification. Again, we employ *censored production rules*, which are production rules with exceptions. This leads us to forms of representation that we believe are more natural and comprehensible than other logically equivalent forms.

The following sections define the meaning and formal properties of censored production rules and show how such rules can be used and learned.

Censored Rules are If-Then-Unless Rules

Each rule in a production system represents a packet of knowledge that is easy to interpret, to explain, and to modify. In this paper, we write production rules in the form:

$$\begin{array}{ll} \textit{If} & \langle \textit{premise} \rangle \\ \textit{then} & \langle \textit{action} \rangle \end{array} \qquad (1)$$

The ⟨*premise*⟩ is a logical product of predicates representing some elementary conditions, and the ⟨*action*⟩ is what is to be done when ⟨premise⟩ is satisfied.

If the ⟨*premise*⟩ part of the rule is not satisfied, no ⟨action⟩ is performed. If the ⟨*action*⟩ part is replaced by a predicate or a conjunction of predicates, then the rule becomes an implicative assertion:

$$\langle \textit{premise} \rangle \Rightarrow \langle \textit{decision} \rangle \qquad (2)$$

Winston [1983] introduced the concept of a production rule augmented with an *unless* condition:

$$\begin{array}{ll} \textit{If} & \langle \textit{premise} \rangle \\ \textit{then} & \langle \textit{decision} \rangle \\ \textit{unless} & \langle \textit{censor} \rangle \end{array} \qquad (3)$$

The $\langle censor \rangle$ is a logical condition (typically, a predicate or the disjunction of predicates) that, when satisfied, blocks the rule. Thus, a censor can be viewed as a statement of exceptions to the rule.

In the original formulation, given in Winston [1983], the censor is logically interpreted according to

$$\langle premise \rangle \,\&\, \neg \langle censor \rangle \Rightarrow \langle decision \rangle \tag{4}$$

which is logically equivalent to

$$\langle premise \rangle \Rightarrow \langle decision \rangle \vee \langle censor \rangle \tag{4a}$$

In this formulation, the role of the *unless* condition is similar to the definition of an *exception* described by Etherington and Reiter [1983]. The difference is that in Winston's formulation there is an additional stipulation that an unlimited effort is put into showing that $\langle premise \rangle$ is true, but only one-step effort is put into showing that $\langle censor \rangle$ is true, and when one-step effort fails, the $\langle censor \rangle$ condition is assumed to be false.

In this paper we present another interpretation of the *unless* condition, discuss its validity, and argue for the utility of rules with the new type of unless conditions.

We believe that our censored production rules capture certain aspects of commonsense knowledge that are absent from ordinary production rules, thereby facilitating human rule creation and comprehension. Our intention is not to develop a cognitive model of human reasoning, however. Consequently, the use of unless conditions in censored production rules is not a precise model of the human use of the word *unless.*

We now treat the logical aspects of unless conditions condition. Once that is done, we treat the expositive and control aspects.

The Unless Operator Is Logically Equivalent to Exclusive-or

Let us consider a simple statement with an *unless* condition: "If it is Sunday, John will work in his yard, unless the weather is bad." Writing this statement as an if-then rule, we have the following:

If	it is Sunday	
then	John works in the yard	
unless	the weather is bad	(5)

If we substitute the propositional symbol S for "it is Sunday," Y for "John will work in his yard," B for "the weather is bad," and denote *unless* by the symbol $\lfloor$, then we can say S implies Y unless B, which we can write as follows:

$$S \Rightarrow Y \lfloor B \tag{6}$$

Suppose we interpret (6) according to (4). Then we write this:

$$S \& \neg B \Rightarrow Y \tag{7}$$

According to (4a), we can also write this:

$$S \Rightarrow Y \vee B \tag{7a}$$

If the weather is not bad, then $\neg B$ is true. And if $\neg B$ is true and it is Sunday, then we can infer that John is in the yard. If the weather is bad, then B is true, the if-part of rule (7) is not satisfied, and nothing can be inferred about whether John is or is not in the yard.

The commonsense meaning of the expression (5), however, supports the inference that if the weather on Sunday is bad, then John does not work in the yard. Such an interpretation of expression (5) requires the following pair of assertions:

$$S \& \neg B \Rightarrow Y \tag{8}$$

$$S \& B \Rightarrow \neg Y \tag{9}$$

These assertions can be writtten equivalently as:

$$S \Rightarrow ((\neg B \Rightarrow Y) \& (B \Rightarrow \neg Y)) \tag{10}$$

By manipulating the *then* part, we obtain

$$S \Rightarrow ((Y \& \neg B) \vee (\neg Y \& B)) \tag{11}$$

and finally,

$$S \Rightarrow (Y \otimes B) \tag{12}$$

where $\otimes$ denotes the exclusive-or operator.

Thus, the logical interpretation of the unless operator, $\lfloor$, requires it to act like the exclusive-or operator, $\otimes$, that connects the *then* part and the *unless* part of a censored rule.

Our unless operator takes precedence over the implication $\Rightarrow$ operator. This new interpretation of the *unless* operator is identical to that of *except for* operator in variable-valued logic defined by Michalski [1980].

Comparing the above described two interpretations of a censored rule, the *passive* one given by equation (7a), and the *active* one given by equation (12), it is clear that one uses the ordinary *or* operator in the right hand side, whereas the other uses the exclusive-or operator. These two interpretations are illustrated graphically in figure 1.

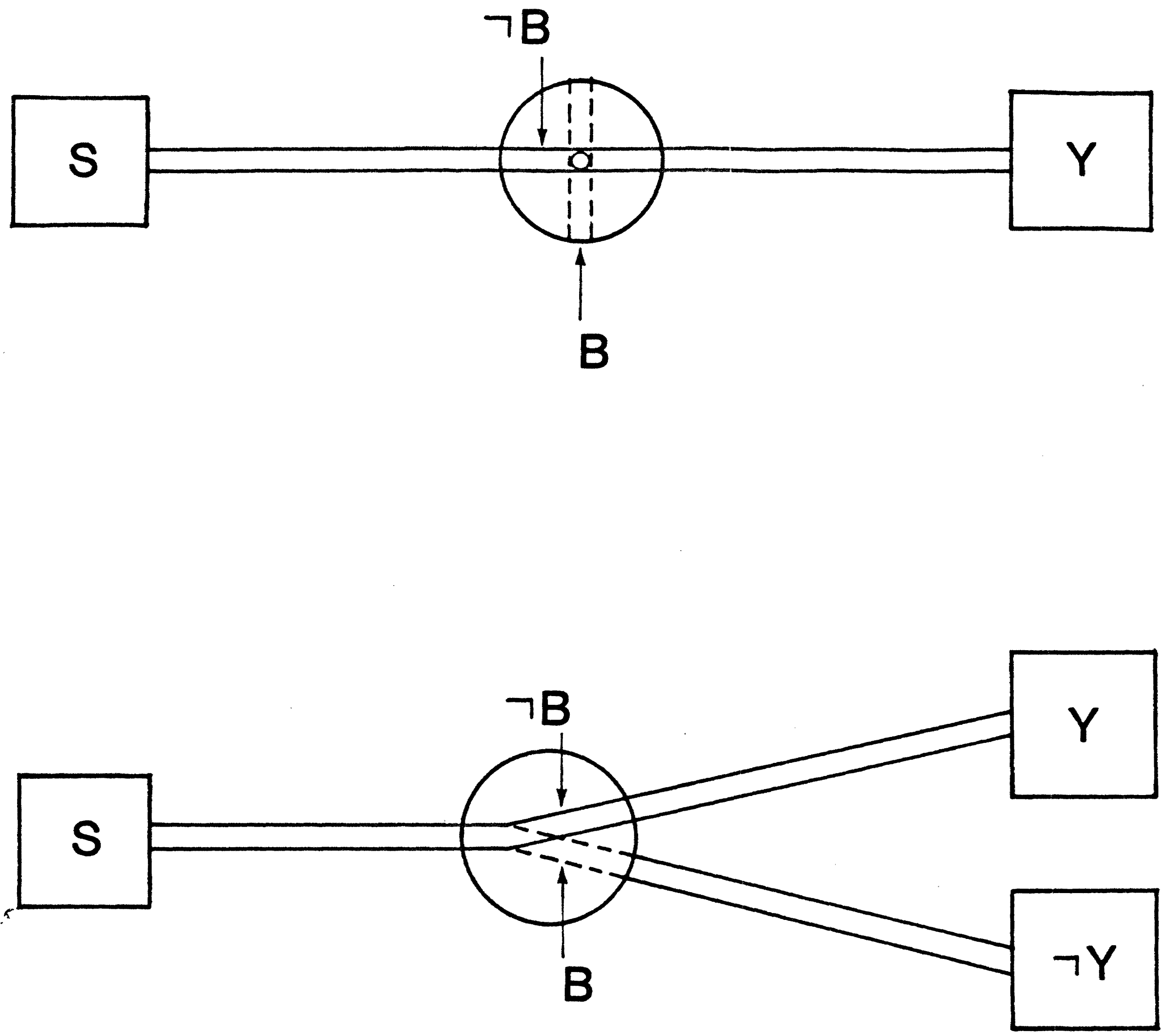

Figure 1.

The Unless Operator Makes Expectations Explicit

Censored rules are important because their unless conditions have extra-logical aspects:

- The unless operator has an *expositive* aspect because it allows us to express certain expectations.
- The unless operator has a *control* aspect because it allows us to deploy a variety of problem-solving schemes.
- The expositive and control aspects of the unless operator constitute its

pragmatics.

In this section, we look at the unless operator from the expositive point of view. In the next, we look from the control point of view.

According to the logical interpretation, the rule

$$S \Rightarrow Y \lfloor B \tag{13}$$

is logically equivalent to

$$S \Rightarrow B \lfloor Y \tag{14}$$

and also to these negation-containing expressions:

$$S \Rightarrow \neg Y \lfloor \neg B \tag{15}$$

$$S \Rightarrow \neg B \lfloor \neg Y \tag{16}$$

Thus, the rule:

If	it is Sunday	
then	John works in the yard	
unless	weather is bad	(17)

should be logically equivalent to the following alternatives:

If	it is Sunday	
then	the weather is bad	
unless	John works in the yard	(18)

If	it is Sunday	
then	John does not work in the yard	
unless	weather is not bad	(19)

If	it is Sunday	
then	the weather is not bad	
unless	John does not work in the yard	(20)

Conversion of (17) to (19), and (18) to (20), is done by negating both the decision and censor conditions. Conversion of (17) to (18), and (19) to (20) is done by swapping the decision and censor conditions.

At this point, we want to focus on the effect of swapping the decision and censor conditions. Let us therefore look more carefully at (17) and (18). These two rules seem to have different meaning. The first rule tells us when John works in the yard, and the second, when the weather is bad. They both imply the same logical conclusion: either John works in the yard and weather is not bad, or John does not work in the yard and weather is bad.

Nevertheless the two rules seem different, because we treat the *unless* condition like a causal precondition for the relationship between *if* and *then* parts. The *unless* condition defines an exception for the relationship. In the example, we know that working in the yard has no influence on the weather. Therefore, the first rule, (17), sounds reasonable, but the second, (18), sounds strange.

Now recall our interpretation of a censored rule, given by expression (10), which we repeat here for convenience:

$$S \Rightarrow ((\neg B \Rightarrow Y \ \& \ B \Rightarrow \neg Y)) \tag{21}$$

A logically equivalent alternative, based on the strange-sounding form, (18), is:

$$S \Rightarrow (\neg Y \Rightarrow B \ \& \ Y \Rightarrow \neg B) \tag{22}$$

The expression (22) provides perspective. Two assertions are implied: "If it is Sunday and John does not work in the yard, then the weather is bad" and "If it is Sunday and John works in the yard, then the weather is not bad." These assertions express the reverse of our sense of the dependence between bad weather and working in the yard.

Censored production rules can be used to represent relationships that involve mutual exclusion, rather than causal dependency between the decision and the censor. Consider this rule:

If	it is Sunday	
then	John works in the yard	
unless	he reads a book	(23)

Transforming according to (14), we have:

If	it is Sunday	
then	John reads a book	
unless	he works in the yard	(24)

Both rules seem to be reasonable and to represent logically equivalent information. But now let us represent rule (23) as a pair of rules corresponding to expressions (8) and (9):

If	it is Sunday and John does not read a book	
then	he works in the yard	(25)

If	it is Sunday and John reads a book	
then	he does not work in the yard	(26)

The first rule clearly implies that on Sunday John either works in the yard or reads a book. We sense that he cannot do anything else on Sunday. There is another asymmetry between (25) and (26). While rule (25) gives us a specific information about what John does, rule (26) seems to be obvious, because we know that a person cannot read a book and work in the yard simultaneously. But when a person does not read a book, he could be doing many things other than working in the yard.

Now let us represent expression (24) as a pair of rules corresponding to expressions (8) and (9):

If	it is Sunday and John does not work in the yard	
then	he reads a book	(27)

If	it is Sunday and John works in the yard	
then	he does not read a book	(28)

Here again, only the first rule supplies useful information. The second rule is redundant because we know, through commmon sense, that a person cannot read a book and work in the yard at the same time.

This leads us to the following observation: if the *action* and the *censor* in a censored rule are mutually exclusive assertions, then the *active* and the *passsive* interpretation of a censored rule yield the same inferences. That is, the *passive* interpretation, together with a commonsense mutual-exclusion constraint, is equivalent to the *active* interpretation.

Let us now consider another example, taken from Winston [1983].

If	a man is an unmarried adult	
then	he is bachelor	
unless	he cannot be married	(29)

Let us transform this rule to an equivalent one according to the same transformation that leads to (16):

If	a man is an unmarried adult	
then	he is not a bachelor	
unless	he can be married	(30)

Clearly, both rules (29) and (30), are perfectly legitimate statements. Also, in both rules, the censor acts as a "switch," validating or invalidating the *then* part of the rule.

There is, however, an important difference between the two rules: in one the censor condition is ordinarily false, whereas in the other, the censor condition is ordinarily true. Consequently, the following implication ordinarily holds:

If	a man is an unmarried adult	
then	he is a bachelor	(31)

Whereas the following implication ordinarily does not hold:

If	a man is an unmarried adult	
then	he is not a bachelor	(32)

Ignoring the censor in rule (29) does not invalidate the rule completely; such a truncated rule still leads to a valid conclusion most of the time. However, ignoring the censor in rule (30) does invalidate the rule completely. As we explain later, this leads to important expositive considerations in connection with the use censored production rules. We will argue that the preferred form of a censored rule is the one in which the if-then part expresses an important, often holding relationship, whereas the censor represents a rarely occurring exception from the rule. This helps justify Winston's assumption, mentioned earlier, that less effort should be expended to prove the *premise* than to prove the *censor*.

Summarizing, there are two important characteristics be remember for censored rules of the form:

$$\langle premise \rangle \Rightarrow \langle decision \rangle \lfloor \langle censor \rangle \qquad (33)$$

- First, from a logical viewpoint, the *unless* operator $\lfloor$ acts like the exclusive-or operator.
- Second, from an expositive viewpoint, $\langle censor \rangle$ condition acts as a switch that validates or invalidates the $\langle decision \rangle$. Its desired use is when the assertion $\langle premise \rangle \& \langle decision \rangle$ holds frequently, and the assertion $\langle premise \rangle \& \langle censor \rangle$ holds rarely.

The Unless Operator Supports Various Control Schemes

So far, we have discussed the unless operator from two points of view: from the logical point of view, we showed that it is equivalent to the exclusive-or operator; and from the expositive point of view, we showed that it makes certain expectations explicit.

Because expectations are made explicit, there are a number of obvious alternative control schemes for using censored rules. Here are the two extreme possibilities:

- The Harry Truman method: Treat the unless operators as if they were exclusive-or operators, thereby ignoring expectation information. Good in situations for which expectations are unreliable.
- The ask-questions-later method: Ignore all censors. Good in situations for which rapid response is critical.

And of course, there are intermediate possibilities, of which the following are representative:

- The decision-maker method: Assume all censor conditions are false unless already known to be true. Put no effort into showing that a censor condition is true.
- The trusting-skeptic method: Once a premise is established, try to show that censor conditions are true, but fix the depth of rule chaining to some prescribed number of levels or to some prescribed consumption of resources. We call this the trusting-skeptic method because it reflects an assumption that the indicated expectations are solid and that there is little point in putting more than a little effort into overturning those expectations. This is the original method proposed by Winston.
- The stubborn-donkey method: Do not allow situations in which censors are triggered by rules that themselves have censors that are triggered by other rules ad nauseum. Fix the depth of censor chaining to some prescribed number of leves or to some prescribed consumption of resources. We call this the stubborn-donkey method because the number of times a conclusion can be reversed is limited.
- The tapered-search method: Allow any number of levels of censor chaining, but reduce the resources allocated to showing that censors are true in proportion to the depth of chaining.

Importantly, for all these and all similar schemes, tentative answers can be reported as soon as the premise-decision parts of the rule base allow. Then the censor parts of the rules can be pursued according to whatever resource allocation scheme seems best. At any moment, the tentative answer is the best answer possible relative to the chosen resource allocation scheme and the expended resources. In principle, tentative answers may change many times, but those tentative answers probably will not change much in practice.

From this point of view, our control proposals are reminiscent of the progressive-deepening notion developed for chess-playing programs. Progressive deepening enables game-playing programs to produce reasonable moves quickly, with better moves forthcomming if the clock allows.

From another point of view, our proposals are reminiscent of the work of Carl Hewitt, whose early thesis argued persuasively that logical rules should be augmented with knowledge about appropriate uses [Hewitt, 1972]. In the thesis, the primary distinction was between antecedant and consequent rules.

While on the subject of control, two other ideas spring to mind:

- Why not devise a parallel-processing scheme?
- Why not devise a more refined, quantitative representation of expectation that would support more sophisticated control schemes?

These ideas seem natural to use, but we have not yet thought much about parallel-processing, nor have we devised control schemes exploring quantitative representation. We have developed a quantitative representation, however, which we describe in the next section.

The Augmented Unless Operator Makes Expectations Quantitative

Let us now give a more quantitative definition of a censored rule. Consider the following rule:

$$P \Rightarrow D \lfloor C \tag{34}$$

where P is a premise, D is a decision, and C is a censor. Although the unless operator $\lfloor$ is logically equivalent to the commutative exclusive-or operator, the unless operator has a expositive aspect which is not commutative. In order to capture the asymetry precisely, let us associate two parameters, γ_1 and γ_2, with rule (34).

$$P \Rightarrow D \lfloor C : \gamma_1, \gamma_2 \tag{31}$$

Both γ_1 and γ_2 are subjective point probabilities, one indicating the strength of the relationship between P and D, and the other, between P and C.[1]

Now consider the following sets:

- Ω is a finite universe of events.
- Ω_P is the set of events for which P holds.
- Ω_{PD} is the subset of events for which both P and D hold.
- Ω_{PC} is a subset of events for which both P and C hold.

[1] It may be better to introduce parameters indicating belief and disbelief for both D and C, as in MYCIN, or to introduce parameters indicating lower bounds on the probabilities of the truth and the falsehood of both D and C, as in INFERNO [Quinlan, 1983]. For simplicity, we consider here only subjective point probabilities.

Given these sets, the parameters γ_1 and γ_2 are defined as follows:

$$\gamma_1 = \frac{|\Omega_{PD}|}{|\Omega_P|} \tag{32}$$

$$\gamma_2 = \frac{|\Omega_{PC}|}{|\Omega_P|} \tag{33}$$

where $|\Omega_i|$ denotes the cardinality of Ω_i.

Relating these definitions to our example about John working in the yard unless weather is bad, we can say that Ω is a set of days over a sufficiently large period of time; Ω_P is the set of Sundays during this period of time; Ω_{PD} is the set of Sundays when John works in the yard; and Ω_{PC} is the set of Sundays with bad weather.

Assuming that there are significantly more Sundays when John works in the yard than there are Sundays when the weather is bad, then Ω_{PD} is considerably larger than the set Ω_{PC}:

$$|\Omega_{PD}| \gg |\Omega_{PC}| \tag{34}$$

Thus, taking into consideration (32) and (33), we have

$$\gamma_1 \gg \gamma_2 \tag{35}$$

In our example, γ_1 stands for the ratio of Sundays when John worked in the yard to all Sundays, and γ_2 stands for the ratio of Sundays with bad weather to all Sundays.

And from the logical point of view, according to our interpretation of the unless operator, the sets Ω_{PD} and Ω_{PC} must be disjoint. Consequently, the sum $\gamma_1 + \gamma_2$, must always equal 1. Thus, knowing γ_1, it is easy to compute γ_2; therefore rule (31) can be simplified:

$$P \Rightarrow D \lfloor C : \gamma \tag{36}$$

where γ stands for γ_1 and $\gamma \geq 0.5$. Because of (35), $\gamma = \gamma_1$ should be significantly greater than 0.5, but we will only assume $\gamma \geq 0.5$. If $\gamma = 1$, then $P\&C$ never holds, and rule (36) becomes

$$P \Rightarrow D$$

.

Note that γ does not say anything about how often relations $\neg P\&D$ and $\neg P\&\neg D$ hold.

Now suppose that γ is 0.9 in the following rule:

$$P \Rightarrow D \lfloor C : \gamma \tag{37}$$

If P holds, rule (37) allows us to make inference that D and $\neg C$ hold with certainty 0.9 and $\neg D$ and C will certainly 0.1. Thus if we know P holds and do not know whether C holds or not, we infer that D holds with certainty 0.9. On the other hand, if P holds and C does not, than we can infer that D holds with certainty 1.

If both P and C hold, then we can infer that D does not hold. Symetrically,, if both P and D hold, we can infer that C does not hold.

Now suppose that we know D but know neither P nor C. If we ignores γ, then (37) can be written as a pair of expressions:

$$P\&\neg C \Rightarrow D \tag{38}$$

$$P\&C \Rightarrow \neg D \tag{39}$$

Expressions (38) and (39) can be rewritten as follows:

$$\neg D \Rightarrow \neg P \vee C \tag{40}$$

$$D \Rightarrow \neg P \vee \neg C \tag{41}$$

If $\neg D$ holds, then either $\neg P$ or C. If we know that $\neg C$, then using (40) we can infer with certainty 1.0 that $\neg P$. If C, then nothing can be said about $\neg P$ in this case.

Similarly, if D holds and we know that C, then we can infer with certainty 1.0 that $\neg P$. If $\neg C$, then nothing can be said about $\neg P$.

Thus rule (37) permits us to generate a number of inferences of varying certainty, depending on what is given and what is unknown. Also, there is a natural relationship between the certainty of conclusions and the amount of knowledge available.

The relationship between certainty and knowledge has an important operational consequence. To illustrate, consider two cases:

- P is known to hold and there are insufficient time or space to determine C. A system can infer the conclusion D, with certainty γ.
- P is known to hold and there are sufficient resources to determine C. A system can determine C and subsequently conclude D or $\neg D$, depending on C, with certainty 1.

Rules may have Many Censors

A censored production rule may have more than one exception-denoting censor. Consider, for example, the assertion that birds fly:

$$\forall\ x \text{ is-bird}(x) \Rightarrow \text{flies}(x) \tag{42}$$

This general assertion enables us to expect that any newly observed bird flies. But not all birds fly. For example, penquins, ostriches, emus, kiwis, and domestic turkeys do not fly. To include this information, we write:

$$\begin{aligned}\forall\ x \text{ is-bird}(x) \Rightarrow \text{flies}(x) \lfloor (&\text{is-penguin}(x) \\ \vee\ &\text{is-ostrich}(x) \\ \vee\ &\text{is-emu}(x) \\ \vee\ &\text{is-kiwi}(x) \\ \vee\ &\text{is-domestic-turkey}(x))\end{aligned} \tag{43}$$

Thus the exceptions are disjunctively linked together as one censor condition. Suppose we generalize these exceptions into one statement for special birds. Then we can write:

$$\forall\ x \text{ is-bird}(x) \Rightarrow \text{flies}(x) \lfloor \text{is-special-bird}(x) \tag{44}$$

But then, the rule is still not entirely correct. Even a flying bird cannot fly when it is dead or sick or has broken wings. Let us characterize all these situations as bird being in an unusual condition. Then we can write:

$$\begin{aligned}\forall\ x \text{ is-bird}(x) \Rightarrow \text{flies}(x) \lfloor (&\text{is-special-bird}(x) \\ \vee\ &\text{is-in-unusual-condition}(x))\end{aligned} \tag{45}$$

where

$$\begin{aligned}\forall\ x \text{ is-special-bird}(x) \Leftarrow\ &\text{is-penguin}(x) \\ \vee\ &\text{is-ostrich}(x) \\ \vee\ &\text{is-emu}(x) \\ \vee\ &\text{is-kiwi}(x) \\ \vee\ &\text{is-domestic-turkey}(x)\end{aligned} \tag{46}$$

and

$$\forall\, x \text{ is-in-unusual-condition}(x) \Leftarrow \text{is-dead}(x) \vee \text{is-sick}(x) \vee \text{has-broken-wings}(x) \quad (47)$$

Now a bird also cannot fly when its legs are stuck in concrete. This case may also be classified as bird in an unusual condition. Thus, to update our knowledge, we need not change our basic rule; we need only extend our definition of unusual condition:

$$\text{is-in-unusual-condition}(x) \Leftarrow \text{is-dead}(x) \vee \text{is-sick}(x) \vee \text{has-broken-wings}(x) \vee \text{has-legs-stuck-in-concrete}(x) \quad (48)$$

Rules May Have Incomplete Censors

Censors are generalized whenever a disjunctive condition is added, as in going from (44) to (45), or from (47) to (48). The generalized censor, (48), fires in additional situations, preventing the rule from asserting the decision. Thus, generalizing a censor specializes a censored rule. Conversely, specializing the censor (up to its complete removal) generalizes a censored rule.

Let us now go back to the rule (42), and augment it by adding the parameter γ:

$$\forall x \text{is-bird} \Rightarrow \text{flies}(x) \lfloor \text{is-special-bird}(x) : \gamma \quad (49)$$

where γ estimates the probability that any given bird flies. In the case that a given bird does not fly, then according to rule (49) the censor is-special-bird(x) must be true. This is at odds with (45), however, because the bird may be not be a special bird, but rather, in an unusual condition.

Instead of generalizing the censor to correct rule (49), let us introduce a parameter δ that adds an additional uncertainty to compensate for the incompleteness of the censor.

$$\forall x \text{is-bird}(x) \Rightarrow \text{flies}(x) \lfloor \text{is-special-bird}(x) : \gamma, \delta \quad (50)$$

Here is the intended meaning:

- γ is the degree of certainty that a bird flies when we do not know whether it is a special bird or not.
- δ is the degree of certainty that the given bird flies when we know that it is not a special bird.

Thus δ accounts for the fact that the censor is incomplete.

Now let us discuss the meaning of parameters γ and δ generally. Suppose we insert into a censored rule a symbol UNK standing for "unknown:"

$$P \Rightarrow D \lfloor (C_1 \vee \text{UNK}) : \gamma, \delta \tag{51}$$

The symbol UNK represents a disjunction of unknown conditions that could block the inference of D from P.

If we know that the condition C_1 does not hold ($\neg C_1$), then the strength of the implication $P \Rightarrow D$, depends on the condition UNK. If none of the UNK conditions hold, then D holds. If any of the UNK conditions hold, then D does not.

Parameter δ is defined to be the degree of certainty that $P \Rightarrow D$ when $\neg C_1$ is true. This is equivalent to the degree of certainty that there is no implicit part of the censor that holds when $\neg C_1$ is true.

Parameter γ is defined as the degree of certainty that $P \Rightarrow D$ when it is not known whether ($C_1 \vee$ UNK) holds. The implication $P \Rightarrow D$ has the degree of certainty 1 when $C_1 \vee$ UNK is known to be false. Thus, the parameter γ is equivalent to the *a priori* degree of certainty that none of the censors hold.

Obviously the *a priori* degree of certainty of $\neg(C_1 \vee \text{UNK})$ must be equal to or smaller than the *a priori* degree of certainty that $\neg$UNK. Therefore, $\gamma \leq \delta$. Note that $\delta = 1$ if it is certain that there are no conditions in the censor other than C_1.

Let us rewrite rule (51) as two rules:

$$P \Rightarrow D \lfloor C^* : \gamma, \delta \tag{52}$$

$$C^* \Leftrightarrow C_1 \vee \text{UNK} \tag{53}$$

Expression (53) can be rewritten as

$$\neg C^* \lfloor (C_1 \vee \text{UNK}) \tag{54}$$

where $\lfloor$ denotes The symmetrical difference. In this form it clearly states that the censor will not fire unless C_1 is true or UNK is true. Suppose now that P holds and we ignore censor C, i.e., we ignore C_1 and UNK, then we can conclude D with the degree of certainty γ. Parameter γ is therefore called the 0-level strength of implication $P \Rightarrow D$ (because no information about the censors is taken into account). If we evaluate C_1, and C_1 is false, then from P we can conclude D with the degree of certainty δ. Parameter δ

is thus called the 1-level strength of implication (the known, m information about the censor is taken into account). Recapitulating our discussion, we define a censored rule as follows:

$$P \Rightarrow D\lfloor C : \gamma, \delta \tag{55}$$

where P is the premise, D is the decision, C is the censor,

γ is the *a priori* degree of certainty that $P \Rightarrow D$ when C is unknown (the 0-level strenght of implication), δ is the *a priori* degree of certainty that $P \Rightarrow D$ when C is known to be false (the 1-level strength of implication).

The Provided Operator Complements the Unless Operator

Consider this statement:

If	It is Saturday	
then	I will go to a concert	
unless	I cannot get a babysitter	(56)

We can express this in our logic as follows:

$$\mathrm{S} \Rightarrow \mathrm{C}\lfloor \mathrm{B} \tag{57}$$

where S is for *on Saturday*; C is for *I will go to a concert*; and B is for *I cannot get a babysitter*.

An alternative way to say the same thing is as follows:

If	It is Saturday	
then	I will go to a concert	
provided	I can get a babysitter	(58)

Similarly, statements such as the following can be expressed using the *provided* form.

You will enjoy the hike *unless* the weather is bad (59)

Happy future is guaranteed *unless* we do not have peace (60)

Changing the polarity of the *censor* condition replaces *unless* by *provided*; the negative censor used with an *unless* condition becomes a positive censor with the *provided* condition, and vice versa.

Of course, in normal human use, the word *provided* introduces a precondition rather than an exception.

Let us now introduce the *provided* operator, denoted $\lceil$, to complement the *unless* operator. Thus, the rule

$$P \Rightarrow D\lfloor C \tag{61}$$

is assumed to be equivalent to

$$P \Rightarrow D\lceil(\neg C) \tag{62}$$

and the rule

$$P \Rightarrow D\lfloor(\neg C) \tag{63}$$

is assumed to be equivalent to

$$P \Rightarrow D\lceil C \tag{64}$$

From the logical standpoint, the exclusive-or operator in $A \oplus \neg B$ is equivalent to $A \equiv B$.

Thus, from the logical viewpoint, the *provided* operator acts as the *equivalence* operator.

Because *unless* and *provided* operators are complementary, any censored rule can be expressed using only a positive censor condition. Also, replacing one operator by another does not effect the strength of implication parameters δ and γ:

$$P \Rightarrow D\lfloor C : \delta, \gamma \tag{65}$$

gaurantees

$$P \Rightarrow D\lceil C' : \delta, \gamma \tag{66}$$

where $C' = \neg C$.

There is a small difference in the rule interpretation: in rule (66), δ is interpreted as the strength of implication $P \Rightarrow D$, when C' *holds.* The parameter γ remains to denote the strength of implication when we do not know if C' holds or not.

Representing the In-which-case Condition

Suppose we want to represent the statement, "On Sunday I will fly a kite, *unless* there is no wind, *in which case* I will write poetry."

The *unless* operator does not enable us to express this statement directly. We could introduce an *in which case* operator, but we prefer to use two statements instead: "On Sunday I will fly a kite *provided* there is wind," and "On Sunday I will write poetry *provided* there is no wind." These statements are directly expressible as censored production rules:

$$S \Rightarrow K\lceil W \tag{67}$$

$$S \Rightarrow P\lceil \neg W \tag{68}$$

Note that (68) can be expressed as a rule with an *unless* condition:

$$S \Rightarrow P\lfloor W \tag{69}$$

The English form sounds strange, however, in light of (67): "On Sunday I will write poetry *unless* there is wind". Rule (68), with a negative censor, seems more appealing than rule (69) with a positive censor. There seems to be a general regularity: from the expositive viewpoint, it is better to use complementary censor conditions and the same censor operator, rather than than to use complementary operators and the same censor condition.

One may ask why to use the censored rules at all, and instead use the ordinary production rules:

$$S \& W \Rightarrow K \tag{70}$$

$$S \& \neg W \Rightarrow P \tag{71}$$

The answer depends on what we want to express. These two pairs of rules are not exactly equivalent. The first pair (rules (67) and (68) can be reexpressed as one expression:

$$S \Rightarrow K\lceil W \;\vee\; P\lceil \neg W \tag{72}$$

This is turn can be rewritten as

$$S \Rightarrow W\&K \vee \neg W\&P \vee \neg(K \vee P) \tag{73}$$

The second pair (rules (70) and (71)) can be reexpresed as

$$S \Rightarrow (W \Rightarrow K)\&(\neg W \Rightarrow P) \tag{74}$$

and then as

$$S \Rightarrow W\&K \vee \neg W\&P \vee K\&P \tag{75}$$

Thus, the diference between the two pairs of rules, (67, 69) and (70, 71), is in the third component of expressions (73) and (75). While the first pair implies that a person may neither fly a kite nor write poetry on Sunday, as well as that he will not simultanously fly a kite and write poetry. The second pair implies that he must do one thing or the other and accepts the possibility that both may be done at once. (This is not to say that flying a kite and writing poetry is mutually exclusive, but only that there is a difference between these two pairs of rules!).

Distinguishing Rules from Definitions

Consider the following expression, which represents the sentence, "On Sunday John will fly a kite *provided* there is wind:"

$$S \Rightarrow K \lfloor W \tag{76}$$

Suppose we also want to express some knowledge about whether there will be wind on Sunday. Assume that this condition is: "There will be wind on Sunday (W) *if* there is drop of temperature on Friday (DF) ***unless*** Saturday is sunny (SS). We can write this condition as a production rule:

$$DF \Rightarrow W \lfloor SA \tag{77}$$

Rule (76) is typically used in the forward direction for answering the question "What will John do on Sunday?" On the other hand, rule (77) is typically used in the backward direction for answering the question "Will there be wind on Sunday?" Rules evoked in the backward direction are called definitions. To reflect the differences in control of rule execution, as well to facilitate human readibility, it is desirable to make a distinction in the form of a forward-executed rule and a backward-executed definition (though logically they are equivalent). Such a distinction can be very simply by writing the rule (77) in the following form:

$$W \Leftarrow DF \lfloor SS \tag{78}$$

This is to be read as W if DF unless SS. The $\Leftarrow$ is interpreted as a logical implication, but directed in the opposite direction than normal. We used this form already in expressions (46) and (47), but without the ***unless*** condition. Note that in this case, in order to interpret the rule correctly, the censor SS is assumed to be linked by the exclusive-or operator with the *left side* of $\Leftarrow$ rather than with the ***right side.***

Note that the operator linking a term being defined with the body of the definition is typically equivalence. Thus, to have a complete logical representation of a definition, one should use the equivalence sign rather than implication.

Inference Rules for Transforming Censored Production Rules

This section gives a sample of inference rules applicable to censored production rules. These rules have one or more premise rules together with an assertion rule that is a logical consequence of the premise rules. To state that an assertion **A** has truth value α we write

$$\mathbf{A} : \alpha \tag{79}$$

To express an inference rule stating that **A** is a logical consequence of $\mathbf{A}_1, \mathbf{A}_2, \ldots$, we write

$$\left.\begin{array}{l} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \cdot \\ \cdot \end{array}\right| > \quad \mathbf{A} \tag{80}$$

We will drop the certainty parameters δ and γ in censored rules whenever they are irrelevant for the given inference rule.

$$\left.\begin{array}{l} P \Rightarrow D\lfloor C : \gamma, \delta \\ P : \text{True} \\ C : \text{Unknown} \end{array}\right| > \quad D : \gamma \tag{81}$$

$$\left.\begin{array}{l} P \Rightarrow D\lfloor C : \gamma, \delta \\ C : \text{False} \end{array}\right| > \quad P \Rightarrow D : \gamma \tag{82}$$

$$\left.\begin{array}{l} P \Rightarrow D\lfloor C : \gamma, \delta \\ D : \text{True} \end{array}\right| > \quad P \Rightarrow \neg C : \text{True} \tag{83}$$

$$\left.\begin{array}{l} P_1 \Rightarrow D\lfloor C \\ P_2 \Rightarrow D\lfloor \end{array}\right| > \quad P_1 \vee P_2 \Rightarrow D\lfloor C \tag{84}$$

$$\left.\begin{array}{l} P \Rightarrow D_1\lfloor C \\ P \Rightarrow D_2\lfloor C \end{array}\right| > \quad P \Rightarrow D_1 \& D_2\lfloor C \tag{85}$$

$$\left.\begin{array}{l} P \Rightarrow D\lfloor C_1 \\ P \Rightarrow D\lfloor C_2 \end{array}\right| > \quad P \Rightarrow D\lfloor (C_1 \vee C_2) \tag{86}$$

$$\left.\begin{array}{l} P \Rightarrow D\lfloor C_1 \\ C_1\lfloor C_2 \end{array}\right| > \quad P \Rightarrow D\lceil C_2 \tag{87}$$

$$\left.\begin{array}{l} P \Rightarrow D\lfloor C_1 \\ D \Rightarrow D_1\lfloor C_2 \\ D_1 \Rightarrow D \end{array}\right| > \quad P \Rightarrow D_1\lfloor (C_1 \vee C_2) \tag{88}$$

In rule (88), the implication $D_1 \Rightarrow D$ is needed to prevent the possibility of having C_1 and D_1 hold simultaneously.

For illustration of (88), let us consider an example. Suppose that the input assertions are

On Sundays John Brite goes to a park unless he is writing poems (89)

In the park he flies a kite unless there is no wind (90)

He flies a kite only in the park (91)

The inference rule (88) allows us to make the following deduction:

On Sundays John Brite flies a kite (92)
unless he is writing poems or there is no wind

Variable Precision Logic is Related to Non-Monotonic Logic

As mentioned earlier, conclussions from censored production rules depend on the truth-status of the censors. The censors are assumed to be low-likelihood assertions. Therefore, if their status is unknown and one wants to (or has to) spare time or resources for determining it, the censors can be ignored, and the conclussion still has a high likelyhood. These conclussions may have to be revised, if the censors are found later to hold. Thus, the mechanism of censored production rules enables one to make revisions in once accepted conclussions.

This reminds one the non-monotonic logic, which specifically investigates problems of revising beliefs and modifying tentative knowledge. In non-monotonic logic, the basic rule of inference is "If a negation of a formula is not derivable from axioms by inference rules of the first order theory, then accept the fomula as true". To fomalize such a rule, this logic extends the classical logic by introducing a proposition-forming modality M. A proposition Mp states the p is consistent with everything believed. A comprehensive treatment of various theoretical aspects of non-monotonic logic, such as consistency and provability is given in [McDermott and Doyle, 1980].

Such an approach is quite different from the one taken in variable precision logic. We do not introduce any new modalities, but rather a new operator (the unless operator). Also, unlike statements in non-monotonic logic, censored production rules are assigned certainty parameters. These parameters can be used for controlling the execution of the rules, and estimating the certainty of conlusions in any act of inference. Thus, the variable precision logic adopts some aspects many-valued logic.

Summarizing, the goals and methods of variable precision logic are different from that of non-monotonic logic. While tha latter investigates the formal implications from reasoning involving uncertain assumptions, variable precision logic atempts to develop mechanisms for representing and conducting reasoning that reflects different trade-offs between the certainty (and/or specificity) of conlusions and the computational resources needed to derive them.

Conclusion

Classical logic was conceived originally as a prescriptive theory of how an ideal mind might reason. For any inference, it assumes that all needed premises are known in advance, and it assumes that the truth value of those premises do not change. It ignores the time and memory resources needed for reasoning. In the real world, however, both humans and computers often must reason using insufficient, incomplete, or tentative premises. Moreover, both are subject to constraining time and memory limitations.

Nevertheless, both humans and computers must be able to react promtly to new information, and they must be able to change or repair their knowledge when new information produces contradictions or when initial assumptions are withdrawn.

Here we have focused on one aspect of the problem by describing a simple knowledge representation and a reasoning system that enables trade-offs between the certainties of various conclusions and the effort needed to derive those conclusions.

We showed that by factoring out conditions that have low likelihood and by treating them outside of the main line of reasoning, a system can easily exercise different control schemas for rule execution.

Commonsense reasoning seems to follow the most important and likely lines of argument, ignoring myriads of low-likelihood exceptions. Variable-precision logic, through the mechanism of censored production rules, provides a simple computational mechanism for capturing some of the properties of such reasoning. The same censored-production-rule mechnamism also facilitates minor repairs to the rules.

We also suggest that a computationally-limited reasoning system should associate rule premises and censors with numerical estimates of their likelihood and testing cost. These estimates enable an inference system to decide which premises and censors to evaluate under given time and cost constaints.

The control planning problem is one of several important topics that were not discussed here, but which clearly invite further research. For example, we need to understand how a reasoning mechanism can make controllable trade-offs between certainty and specificity.

We also need to devise efficient algorithms for learning censored production rules from specific cases and precedents, and to modify those rules incrementally to account for new facts. Some initial work in this direction was recently done by Becker (1985).

References

Becker, J. M., Inductive Learning of Decision Rules with Exceptions: Methodology and Experimentation, M.S. Thesis, Department of Computer Science, University of Illinois, Urbana, IL, July, 1985.

Etherington, David W., and Reiter, Raymond, "On Inheritance Hierarchies with Exceptions," *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, August 22-26, 1983.

Hewitt, C.E., Description and Theoretical Analysis (using schemata) of PLANNER: a Language for Proving Theorems and Manipulating Models in a Robot, MIT AI Laboratory, TR-258, 1972.

McDermott, Drew, and Doyle, John, "Non-Monotonic Logic I," *Artificial Intelligence*, vol.13, No. 1,2, 1980.

Michalski, Ryszard S., "A Theory and Methodology of Inductive Learning," in *Machine Learning—An Artificial Intelligence Approach*, edited by Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell, Tioga Publishing Co., Palo Alto, CA, 1983.

Minsky, Marvin, "The Society of Mind", forthcoming.

Reinke, R., and Michalski, Ryszard S., "Incremental Learning of Concept Descriptions, *Machine Intelligence 11*, J. E. Hayes, D. Michie, and J. Richards, eds., Oxford: Oxford University Press, 1985 (in press).

Quinlan, J.R., Inferno: A Cautious Approach to Uncertain Inference, *The Computer Journal*, vol. 26, No 3, pp.255-269, 1983.

Winston, Patrick Henry, "Learning New Principles from Precedents and Exercises," *Artificial Intelligence*, vol.19, No.3, 1981.

Winston, Patrick Henry, "Learning by Augmenting Rules and Accumulating Censors," *Proceedings of the International Workshop on Machine Learning*, Allerton House, University of Illinois, June 22-24, 1983 (an updated version will appear in *Machine Learning: An Artificial Intelligence Approach Volume II*, edited by Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell, in press.)