# Dynamic Modification of Expert System Production Rules Through Activation Sets

Zofia A Roberts
ZML Software Systems, Louisville, Kentucky, U.S.A.



A production rule based inferential structure of an expert system defines a set of relations on the states of its transient, domain-dependent knowledge. If the inferential structure is heuristic, then it is necessary to allow the set of relations to be redefined, i. e., modifications of the inferential structure may be required.

Such modifications of the inferential structure can be broadly divided into two kinds:

(1) static modifications, and
(2) dynamic modifications.

We identified three types of static rule changes:

(1) addition of rules,
(2) deletion of rules, and
(3) modification of rules.

They involve external changes done by the designers/programmers of an expert system and do not concern us here. This paper explores the possibilities of changing the production rules dynamically. We propose that the changes are classified as:

(1) dynamic activation of rules,
(2) dynamic de-activation of rules, and
(3) dynamic modification of rules.

In order for an expert system to dynamically change its inferential structure, its control structure must be explicitly implemented and equipped with meta-knowledge about the inferential structure. We have suggested elsewhere (Roberts 1986) that such knowledge can take a form of *activation sets*, i.e., a set of conditions which define the applicability of a production rule, or a set of production rules, to the "input" of the inferential structure. The "input" to the inferential structure is a subset of the transient, domain-dependent knowledge of an expert system (Bandler 1985), and was termed a set of *transinferents* (Roberts 1986) for short. For instance, in a crisp, modus ponens implication used as a production rule:

| | | |
|---|---|---|
| (1) "IF a AND b THEN c" | | |
| (2) "a" | ⟵ | transinferent |
| (3) "b" | ⟵ | transinferent |
| conclude: "c" | ⟵ | transinferent |

the activation set for (1) above can either consist of "a AND b", or a, b. The meta-knowledge, i. e., the activation sets can be represented as relational tables. The rows of such a table are production rules *labels*, and columns - *activation sets.* Consider, for example, an inferential structure which consists of the following rules:

rule 1: IF a AND b THEN conclusion 1
rule 2: IF a AND c THEN conclusion 2
rule 3: IF b AND d THEN conclusion 3

The concept of activation sets is general enough to allow great flexibility in the choice of desired control resolution level. For the above set of rules the finest resolution level will treat the set a, b, c, d as the basic control "input", in which case the table will take the following form:

| | a | b | c | d |
|---|---|---|---|---|
| rule 1 | 1 | 1 | 0 | 0 |
| rule 2 | 1 | 0 | 1 | 0 |
| rule 3 | 0 | 1 | 0 | 1 |

Activation Table 1

If coarser resolution is required, then the basic control input may be defined at the level of (a AND b), (a AND c), (b AND d) which results in the following table:

| | a AND b | a AND c | b AND d |
|---|---|---|---|
| rule 1 | 1 | 0 | 0 |
| rule 2 | 0 | 1 | 0 |
| rule 3 | 0 | 0 | 1 |

Activation Table 2

In the above tables the "cell" values are crisp, 0, 1, and may be used to represent only two facts:
(1) the activation set (column heading) participates in a rule;
(2) the rule does not exist.

Incidentally, at the finest resolution level (Activation Table 1) the column headings represent transinferents, while column headings of Activation Table 2 represent sets of transinferents which can be interpreted as "contexualized input". That is, the above set of rules (rules 1 - 3) is only sensitive to the transinferent "a" if it occurs in the context of the transinferents "b" or "c".

The above scheme of representing control knowledge allows to de-activate rules quite easily by changing a "1" in a table cell into a "0". If the system is equipped with a learning module, then the control of an expert system is capable of detecting the absence of a rule and could invoke a learning module to acquire it. For instance, consider activation table 2 above, again. If "a AND d" is detected in the transient, domain-dependent knowledge, since the control finds no heading for it in a table, it activates a learning module.

However, a crisp model has a number of disadvantages. It does not allow rule modification, since the meta-knowledge consists basically of two facts: (1) the rule exists, and (2) the rule does not exist.

To even distinguish between non-existence and de-activation of rules one needs to extend a model to three valued logic. Bandler and Kohout (1985, p. 347) define a *graded production rule* as a rule which uses "grades or degrees of strength of implication, leading to grades or degrees of certainty or possibility attached to the conclusion." They give an example of modus ponens in multi-valued logic (MVL):

| crisp modus ponens | MVL modus ponens |
|---|---|
| $a \rightarrow b$ | $a \rightarrow b = x$ |
| $a$ | $a = y$ |
| ——— | ——— |
| $b$ | $b := z$ |

and they note that "this offers the valuable opportunity to modify the production rules during operation." (Bandler and Kohout 1985, p. 352) They propose a following scheme:

MVL modus ponens

$$\begin{array}{l} a \rightarrow b = x \\ a = y \\ \hline a \rightarrow b := x' \\ a := y' \end{array}$$

"Here an inconsistency of data, or between data and knowledge, is revealed in the incompatible values x and y, and the system has the opportunity to revise these values so as to accord with possibility. Ordinarily, re-evaluation of "y" would be an alteration of data and re-evaluation of "x,' a modification of a systematic "belief" or procedure." [Bandler and Kohout (1985), p. 352]

It is a trivial extension to allow meta knowledge to be graded in their sense. The cells of the Activation Tables 1 and 2 will simply be allowed to hold values from the fuzzy interval [0, 1], instead of the crisp 0, 1 set. These fuzzy values are then manipulated by the controller of an expert system.

The advantages of this extension are enormous. Dynamic modifications of production rules are now a matter of changing a cell value. The inferential structure may then be "flagged" as modified, and would need to "look up" the new values. These modifications can be stored as control experience of an expert system, and strategies as well as analyses can be developed based on it. For instance, sequences of rule modifications may reveal trends in data.

Thresholds can be established below which rules are considered "de-activated". A de-activated rule may not be allowed to participate in the reasoning process.

The number of activations can be counted and rules which are not invoked may be removed from the system. Consistency checks are made easy. Such a module could check the activation tables to detect incompatible and/or inconsistent rules. Redundant rules can be removed in a similar manner.

If an expert system is goal-directed, then its goals can be defined in terms of the activation set tables. Activation set tables can be easily implemented. Two schemes are possible: either each production rule can be required to initially declare its identity, activation sets and cell values, and based on this information the control of an expert system will build the tables; or the control of an expert system can "learn" of them dynamically if we allow the system to initially be totally data-driven for some arbitrary period of time. During this first phase the activation tables can be created by the production rules themselves. After the initial stage the control would be returned to its proper agent, the controller of an expert system.

We have thus shown that a combination of Bandler and Kohout's concepts of gradedness of production rules and the notion of activation sets allow dynamic modifications of production rules of an expert system and deem the control of such a system more versatile, its inferential structure easy to change and its processing more efficient, we hope. Of course, the activation set tables require an implementation to test their validity and true utility.

# References

Bandler, Wyllis. 1985.
Representation and manipulation of knowledge in fuzzy expert systems. *Int. J. Systems Research and Information Sciences* **1**, pp. 113-116.

Bandler, Wyllis and Ladislav J. Kohout. 1986.
Mathematical relations. In M. G. Sigh et al (eds.) *International Encyclopedia of Systems and Control.* New York: Pergamon Press.

Bandler, Wyllis and Ladislav J. Kohout (1985)
Probabilistic vs.fuzzy production rules in expert systems. *Int. J. Man-Machine Studies*, **22**, pp. 347-353.

Kohout, Ladislav J. (1988)
*A Perspectives on intelligent systems: A framework for analysis and design.* London, Chapmann and Hall.

Roberts, Zofia A. (1986)
The Functional Structure of Control in Expert Systems. *M. S. Thesis*, Department of Computer Science, Florida State University, Tallahassee, Florida.