Symbolic Knowledge Processing for the Acquisition of Expert Behavior: A Study in Medicine

Alain T. Rappaport*

Jean-Marie C. Chauvet**

CMU-RI-TR-84-8

Robotics Institute
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

*Faculté de Médecine Necker-Enfants Malades, Paris
INSERM U7, Hopital Necker, Paris

**École Polytechnique, Palaiseau

May 1984

Table of Contents

1. Introduction 1
    1.1. Motivation: simulation of expert behavior acquisition 1
2. Formalization of the system 3
    2.1. Knowledge Sources 3
        2.1.1. Non-alterable knowledge sources 3
        2.1.2. Alterable knowledge sources 3
    2.2. Scope and description of the subsystems 4
        2.2.1. KSI subsystem 4
        2.2.2. NCLOSE subsystem 5
        2.2.3. KAA subsystem 5
    2.3. General Overview 5
    2.4. Organization of this paper 6
3. Knowledge base 6
    3.1. Rule-based knowledge 6
        3.1.1. Nodes: goals and subgoals 7
        3.1.2. Links 7
        3.1.3. Relevance lists 8
        3.1.4. Network organization 8
        3.1.5. Knowledge-base construction 8
    3.2. Plan base 8
    3.3. Results 8
4. Problem-solving module 9
    4.1. Method 10
        4.1.1. Evaluating hypotheses in Medicine 10
        4.1.2. Computational aspects 11
        4.1.3. Object representation 11
        4.1.4. Initial input 11
        4.1.5. Control Structure 12
    4.2. Results: example of a session 16
5. Knowledge Aggregation Algorithm 16
    5.1. Introduction 16
    5.2. Method 17
        5.2.1. Reasoning Pathways representation 17
        5.2.2. Symbolic distance 17
        5.2.3. Symbolic Concept Aggregation 17
        5.2.4. The Concept Knowledge Network 18
    5.3. Results 19
        5.3.1. Processing of the reasoning pathways 19
        5.3.2. Global behavior of the symbolic aggregation algorithm 20
6. KSI: Knowledge Structure Interface 20
    6.1. Method 21
        6.1.1. Symbolic determination of first-look signs - 21
        6.1.2. Determination of relevant clusters 21
        6.1.3. Determination of first-look signs 21
    6.2. Altering hypothesis generation 22
    6.3. General implications 22
7. General Results and Discussion 23
    7.1. Global Results 23
        7.1.1. The learning system 23

7.1.2. The expert behavior 24
7.1.3. Use of first-look signs 26
7.1.4. Medical Interpretation 26
7.2. Instability and Perturbations 27
7.3. Expectations and initial formulation 27
7.4. The alterable rule base 28
7.5. An unstable system 29
7.6. Free-association and task-oriented structures 29
7.7. Prospects 30
7.7.1. Learning methods and future implementation 30
7.7.2. Control Structure Representation Language 30
7.7.3. Patient Evolution 30
7.7.4. Explanatory Module 30
7.7.5. A Network of Knowledge Sources 31
7.7.6. Critical signs and Plan Interaction 31
7.7.7. Modular medical intelligent systems 31
7.8. Summary and Conclusion 31
8. Acknowledgments: 32
I. Problem-Solving example 33
II. Formalization 36
II.1. The symbolic proximity 36
11.2. Transitive Closure of a matrix 37
11.3. Clusters, modules and congruences 37
III. A control structure for rule-matching 38

List of Figures

Figure 2-1:	Modules and flow of information	4
Figure 4-1:	General mechanism of the NCLOSE algorithm	12
Figure 4-2:	Simple model of propagation and control	15
Figure 5-1:	Actual vs Polynomial growth of network size	19
Figure 7-1:	Examples of Clusters	23
Figure 7-2:	Acquisition of first-look generation of hypotheses	25
Figure 7-3:	Instability and Perturbations	28

Abstract

This research is concerned with the simulation of learning by experience to induce the capability for a knowledge-based system to pre-structure the problem before solving it. The model we present is made of different consecutive modules accounting for the tasks of problem-solving, building a dynamic memory and extracting expectations, and pre-structuring or pre-solving the problem. The problem-solver yields internal representations of the problems between which symbolic distances may be defined. The latter are then processed to build the dynamic memory. We used the formalization of medical problem-solving as an example, studying how successive evaluations of cases may lead to the acquisition of the capability to generate an accurate set of initial hypotheses: an expert behavior. The knowledge base is not modified, neither are the strategies in the present implementation. To the data gathering about the patient's complaints is added a concept-driven process by which the system asks for specific data representative of the past experience. The results show that such a system, evolving in a coherent reality increases its qualitative behavior by initially focusing on the right hypotheses or goals. This improvement is induced by the exposure to new situations. Moreover, situations once or rarely encountered are efficiently recognized when re-occurring later.

1. Introduction

> *In the real world it is necessary that doctors not only understand the statistical relations of signs and symptoms to the various possible diseases but also have the wisdom and common sense that derive from the understanding and experience of everyday human existence. It is this last requirement that represents the greatest weakness (and perhaps the ultimate limitation) of computer technology in dealing in any comprehensive fashion with the problem of clinical diagnosis.*[1]

This paper describes a modular expert system in medicine, as a model to study the acquisition of expert behavior from experience, by adaptive learning. It comprises both *task-oriented* and *free-association* methods to account for a *learning by experience*. It is based on an analysis of medical problem-solving, where an important aspect of expert behavior may be the capacity to generate a most accurate set of initial hypotheses, before entering the precise task of problem-solving. Part of this *intuition* or *expectation* is believed to be inferred from experience.

The examples of results presented here show that the system improves its generation of initial hypotheses with experience. This is not done by *repeating* the same cases, but by presenting *new* cases. In order to behave more efficiently with regard to already known situations, the system must in fact meet *different* ones. Moreover, a single occurrence of a different case can be very well recognized even though the second occurrence might take place much later. While such effects have been observed when building an expectation based on *expected* facts, we have also looked at a system expecting the *unexpected.*

The system's modules and mechanisms, all involved in each session, can be summarized as follows:

- A *knowledge-base* containing the most elementary chunks of knowledge in the domain, in the form of rules given by an expert.

- A *problem-solving strategy* for structuring that knowledge, based on the principle of *differential diagnosis.* This module yields *internal representations of the problems* to be further processed. The methodology derives from production systems.

- An *endogenous mechanism* for building *concept knowledge* from the previous outputs. The methodology derives from cluster analysis.

- A search system to *generate adequate hypotheses* from internal conceptualizations and from external data. The methodology derives from Set Theory.

This study involves the problems of changing knowledge representations, applying successively different computational methods without altering meanings and correct control of the flow of information. It provides a model for this type of study, suggesting a different approach to the problem of learning and efficiency of knowledge-based systems.

1.1. Motivation: simulation of expert behavior acquisition

Most expert systems, or intelligent systems, in medicine were built assuming that when functioning they should indeed be experts right away [42, 34, 32].

---

[1] G. Octo Barnett, *The Computer and Clinical Judgment*, New England Journal of Medicine, 1982, 307:493-494

Encoding a behavior that is essentially the result of a long interaction with reality, can indeed be extremely difficult [46]. Learning systems have been studied in Medicine, particularly by processing rules and increasing the quality of the knowledge base [22, 23]. We chose to build a model that would allow us to start studying how expert *behavior* is acquired. This model, using the formalism of rule-based systems for problem-solving, should not improve by means of updating weights based on probabilistic analysis, but rather by modifying the state of a memory where elements of knowledge are semantically linked. The tradeoff should account for simulating the *instability* of human thought, disturbed by a single, significantly unexpected event, but then well remembering this disturbance. This implies a system where *expectations*, drawn from a dynamic-memory, are built and adapt to reality. The use of an incremental rule processor would allow recognition of specific situations but not related ones as well. Such new rules would be triggered during the evaluation process, and do not represent *expectations* resulting in a pre-structuring of the problem.

Medical problem-solving can be formalized into two different consecutive tasks, namely generation of initial hypotheses and evaluation of the latter [16]. While the evaluation process might itself force the evocation or generation of new hypotheses to investigate, generation of the initial set of hypotheses is based on data gathered from the patient's complaints *and* from a set of important and discriminant cues that the physician has learned from experience. These cues, named *first-look signs*, are patient-independent, but experience-dependent. They might be altered by exposure to a series of similar cases or a few very unusual ones and thus represent the physician's state of expectation.

Assuming that a non-expert may benefit from the same basic fact knowledge base as an expert does, the difference between the two in handling a case actually relies on the ability to initially focus on an optimized set of hypotheses, thereby having pre-structured the problem space before starting the evaluation phase.

Evaluating hypotheses requires a strategy for structuring and searching through the knowledge base. This strategy can be taught to the non-expert as part of the knowledge. Although it could also be modified by experience, for instance to build heuristics controlling the *depth* of search, we will assume that it is not in our model since it would not directly affect the initial generation of hypotheses. Rather, we postulate that both experts and non-experts use the same strategy for evaluation, and the same knowledge. This knowledge might be increased *horizontally* by adding new facts, thus giving the expert more knowledge, but it would be the same kind of *book knowledge.* The non-expert or novice possesses *only* this strategy, based on the principle of performing a differential diagnostic task, and *cannot* use qualitative relations between symptoms for he or she has not discovered them *yet.*

Through successive evaluation of cases, the non-expert *acquires* an *interpretation* of reality inducing a previously absent *general expectation.* In effect, the medical expert has a *first-look* capacity of *pre-structuring while approaching* problems, founded on experience. It may be expected from a simulation that the quality of this first-look approach will depend both upon the long term experience and its modifications, and upon recent exposures to unexpected cases.

Although specific, the model of medical reasoning and experience does fit a more general view of *resource* processing in humans:

The human mind is alert to a variety of discrete external data as well as to sets of such data, or *situations.* Moreover, those situations may be highly unexpected. Thus, by analogy with Norman [31], we can postulate two major mechanisms for *resource* processing which apply to physicians:

- A *data-driven* guidance, with an endogenous problem-solving device, responsible for *evaluation* ***tasks.***

- A *concept-driven* guidance, where resources are abstractions resulting from the processing of data from the problem-solver by higher functions. The concept-driven guidance is primarily responsible for the *state of expectation* of the system, as it *arouses* specific slots of the data-driven mechanism. In other words, the concepts issued from the processing units and the memory *modify* the *threshold* of certain data-collecting slots. For the physician, those slots correspond to the first-look signs.

2. Formalization of the system

The expert system we present is made of different functional subsystems involving the use of various approaches and tools in Artificial Intelligence. In the next subsection, these subsystems are briefly described along with their underlying technical formalization.

2.1. Knowledge Sources

The system can access different knowledge sources during execution. These knowledge sources fall into two distinct classes : *alterable* and *non-alterable* sources, with respect to the system itself. Knowledge sources of the first class might be created, altered or deleted by one or more subsystems as opposed to knowledge sources of the second class. These latter knowledge sources, described here, can only be altered by a process of *instruction* relying on interaction with human experts, and not with the system itself.

2.1.1. Non-alterable knowledge sources

The two non-alterable knowledge sources present in the system are:

Rule Base: A production memory where rules or productions appear in the classical condition-action pair format. The LHS and RHS refer to the Patient Object Memory. The syntax of these rules is a simplified LISP representation, using terms intelligible to the physician and related to the domain of application. Additional information is provided for the matching algorithm with a list of relevant signs for each rule acting as a context for the production.

Plan Base: The plan used to guide the control structure of the matching subsystem, described as an instance of a frame or flavor, containing specific slots. Slots specify subsets of items in the Patient Object Memory, ordering them according to clinical considerations. Each of these classes of signs is characterized by its name and rank.

2.1.2. Alterable knowledge sources

Subsystems use alterable knowledge sources for communication purposes. These sources handle communications with the user as well as inter-subsystem communications.

Patient Object Memory: A frame containing a great number of slots, the values of which are signs. This frame is referenced by the Rule Base by means of requests for certain values of theses slots relevant to the rule considered. During the evaluation process, signs are asked the user when needed. The slots are furthermore ordered by the Plan Base. A case is defined by instantiating signs for some slots.

Dynamic Memory: Evolving structures accounting for generalization and learning from cases. We call the content of dynamic memory *concept knowledge*, whose concepts or *clusters* are subsets of reasoning pathways or traces of execution of the evaluation process. Each pathway is associated with a case, and thus refers to a particular instance of signs in the Patient Object Memory. Each pathway is associated with a trace of the evaluation process containing rules in the order they were fired. The organization of these pathways is dynamically modified by the aggregating subsystem. The overall representation of the dynamic memory is a set of different partitions of the current set of pathways indexed by a list of signs. The number and indexes of partitions present in the dynamic memory as well as their structure may change as the system runs. They are internally encoded in LISP lists.

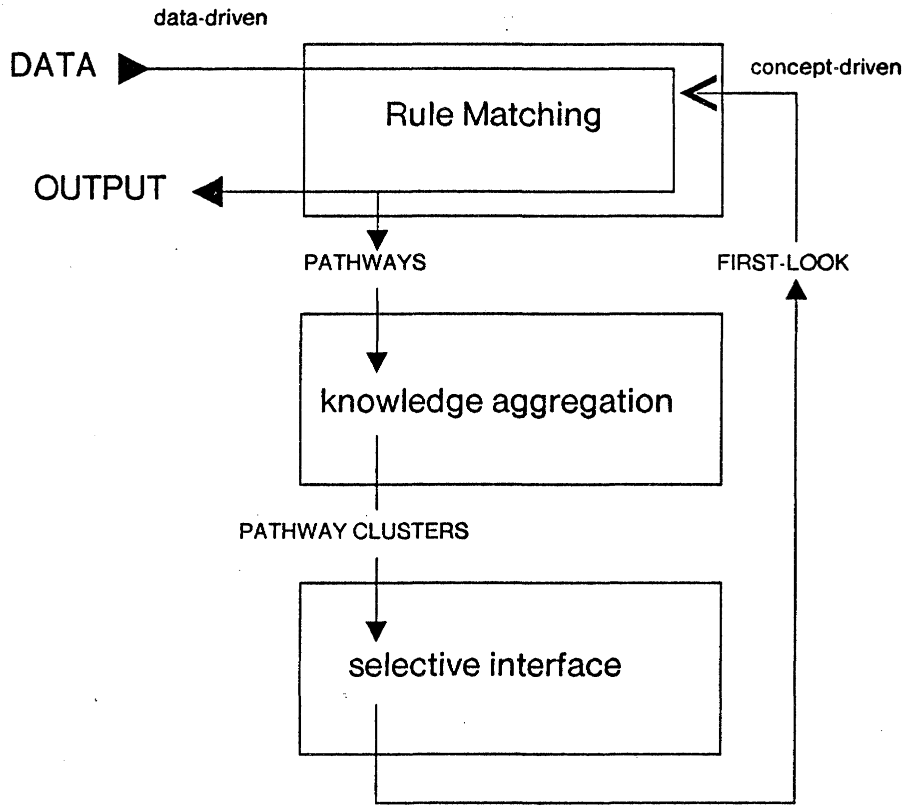

Figure 2-1: Modules and flow of information

2.2. Scope and description of the subsystems

Three subsystems perform distinct operations using the preceding knowledge sources (Figure 2-1). We will use the following notations for those subsystems: KSI, NCLOSE and KAA.

2.2.1. KSI subsystem

KSI, the Knowledge Structure Interface subsystem, performs what we denote as a *first-look* operation.

KSI accesses the Dynamic Memory, retrieving as an input a set of clusters computed by the KAA subsystem. By instantiating particular slots of the Patient Object Memory, KSI yields a list of *first-look hypotheses.*

KSI performs intersection and union operations on the list of signs of the rules appearing in the different paths of the cluster, according to criteria of abstraction and specificity.

2.2.2. NCLOSE subsystem

NCLOSE is a matching and evaluating algorithm. It is a production system with an enhanced recognize-act-cycle accounting for a differential diagnosis control structure.

NCLOSE accesses the Rule Base as a production memory, and the Patient Knowledge Source as a working memory. The output is a reasoning pathway list. NCLOSE accesses the Plan Base at various points in the evaluation process.

The NCLOSE control structure for resolution of the many objects/many patterns problem makes use of differential diagnosis. From a limited number of hypotheses yielded by KSI, NCLOSE performs backward chaining towards the signs present in the working memory with possible request to the user for values of needed slots in the patient frame, and then forward chaining to rules triggered by these signs. As soon as a rule is instantiated, it is fired. The order of rule evaluations is inferred using the Plan Base. This backward forward cycle is iterated until the instantiable rules have all been fired.

An additional subsystem uses discrepancies between the initial hypotheses list and the final hypotheses to modify the choice of the next initial hypotheses, by noticing certain signs responsible for the perceived differences.

2.2.3. KAA subsystem

KAA, the Knowledge Aggregation Algorithm subsystem, is the learning subsystem. Using past experience, i.e. evaluations of different cases, it alters the dynamic memory, improving its own representation of knowledge. KAA builds and uses the Dynamic Memory as a source of knowledge for processing pathways. This processing results in alterations of the organization of the Dynamic Memory.

KAA is an incremental process, accepting reasoning pathways as cases are evaluated. After a clustering analysis, KAA draws expectations about the next input pathway. Differences between actual input and expectations induce modifications of the number and organization of partitions in the Dynamic Memory. The clusters are built by means of a proximity notion between sets of pathways, involving Set Theory. The output is a set of clusters matching the input.

2.3. General Overview

Patient data, collected by volunteered complaints and inquiries about first-look signs, allow the selection of initial hypotheses. The latter are evaluated by a Rule Matching algorithm as shown in figure 2-1. The resulting pathways, i.e. traces of the evaluation process, are used to update the dynamic memory. This is done by an aggregation procedure yielding clusters of such pathways. These clusters are finally used by the Knowledge Structure Interface to select first-look signs. The system has thus acquired an internal representation of the outside world and hence an expectation which affects its behavior towards an *active* relation with the world. This new relation through the system's own perception of events enhances the *passive* observation of a purely data-driven problem solver.

2.4. Organization of this paper

The next section describes the knowledge base formulation we adopted. The structure of the rules is presented along with some particular aspects of the building of the base

Section 4 describes the rule-matching algorithm designed on the basis of some important observations concerning medical reasoning which are succinctly presented. Results obtained by testing this algorithm on actual case records are then described and briefly discussed.

Section 5 describes the knowledge aggregating algorithm, presenting both the numerical and symbolic computation method of distance matrices between outputs of the previous algorithm. A description of the resulting organization of knowledge follows, along with the presentation of actual results.

Section 6 describes the method by which the general expectation, represented by the first-look signs, are inferred from the network of abstractions established by the previous algorithm, and how they are then exploited to generate initial hypotheses and, hence, expert behavior.

Section 7 describes an example of expertise acquisition with this system, and a general discussion.

3. Knowledge base

3.1. Rule-based knowledge

Medicine is an ideal field for the problem of knowledge representation, because the knowledge involved has such a wide span, from universal facts to local trends, and from scientific data to social and psychological problems. Whatever the nature of the arguments involved in medical problem solving, the result always *must be a decision*. In other words, the process as a whole must *converge* towards a solution. This might be a constant rule, as even *not* to decide is to decide. Furthermore, we can assume, by reasoning in a top-down manner, that the elements into which this process can be decomposed are of the same nature and, therefore, may be decisional propositions.

The knowledge to be represented is *basic knowledge* in that it contains the fundamental elements, or building blocks, of the forthcoming experience. It must be understood as the material given to the medical student during the lectures by the *expert* professor. It contains the arguments of different natures that are to be considered when evaluating a problem in the domain. Thus, although it is expert-level knowledge, it does not contain any of the *intuitions* that allow *expert consultation.* There is no rule processor to account for rule modification or making of new rules infered from experience, nor are there probabilities or other numerical weights assigned.

Rules have the following format:

```
(macro
 (list of relevant signs)
 (conditions)
 (hypothesis confirmed and/or object modification))
```

where

- *macro* is the name of the LISP macro function that reads the knowledge file.

- *(list of relevant signs)* is a list of signs used for the propagation by differential diagnoses. It may contain signs which are not comprised in the conditions of the rule's LHS (Left-Hand Side). Thus, it represensents an *evoking context* for the rule.

- *signs* are arguments of various nature, clinical symptoms, laboratory data or any information that might help evoke a particular hypothesis.

- *(conditions) arc* the tests to be validated in order for the rule's RHS (Right Hand Side) to be fired, when the rule is being evaluated. Conditions contain one or more tests that are linked by an *and* logical operator. There is no or logical operator; it is performed by using multiple rules.

- *(hypothesis confirmed and/or object modification)* are the possible action's resulting from the rule's positive evaluation (firing). Each rule is concerned with a single hypothesis. If the rule is fired, the hypothesis is confirmed and the *goal memory* is updated. Modifying the object means modifying the value of one or several of the patient's attribute. The set of the latter constitute the *object memory.*

3.1.1. Nodes: goals and subgoals

Nodes are *hypotheses,* descriptive elements either of the patient's illness or of the physician's actions, according to the problem being solved (diagnostic or therapeutic). They may be classified further into, (i) *goals* which are hypotheses only appearing in Right-Hand-Sides of rules, (ii) *subgoals* which are involved in at least one Left-Hand-Side in the knowledge base. Experts often express their knowledge in such a pre-compiled form. For instance, in order to prescribe an oral contraceptive containing synthetic estrogens the latter must be allowed which means that a large number of conditions must be imperatively met. These conditions are thus assembled in a single rule pointing to the subgoal ESTROGENS-ALLOWED. This subgoal will then appear in the condition (yes estrogens-allowed) belonging to the RHS of a rule pointing to the goal ESTROPROGESTOGENS-NORMAL-DOSES, a final hypothesis.

3.1.2. Links

Links are represented by rules and express a relation between one level of abstraction (signs and/or subgoals) and another (goals and/or subgoals). They are not categorized in a particular way, but may express various types of relations. The latter may be *causal, suggestive* or *constraining.* In the present, study most rules are *suggestiveorconstraining.*

- *Causal links* express a direct cause/effect relation between two facts at any level of abstraction. These links can be established via a RHS action.

- *Suggestive links* express a fact that certain signs and subgoals, when *associated* for any possible reason *argue in favor* of a goal, or a subgoal.

- *Constraining links* result from the verification of many *negative* conditions. A number of rules do contain negative arguments, some only such conditions.

3.1.3. Relevance lists

Each rule possesses a list of relevance signs which locally represents the differential diagnosis operation. Signs belonging to a list are important by their presence, not their value. These signs allow the expansion of the search for differential diagnoses by the control structure. This search, in effect, is based on the possible intersection of causes or associations between two or several hypotheses.

3.1.4. Network organization

The structure of the knowledge base may be viewed as a network of rules with signs and subgoals pointing to goals such as in Figure 4-2.

3.1.5. Knowledge-base construction

We present, in this study, a knowledge-base on Birth Control Prescription Aid[2] (BCPA). The base was built as follows:

A first version was built by a medical student (non-expert situation) using knowledge from previous lectures by various staff members of the expert's department. A revised version was made with the expert. The format of the rules was not modified, and the latter version was tested. Further revisions were made as the rule-based system was tested on actual case records, or day-to-day cases. A version is now available, which represents the views in this domain, of this particular expert. Another expert kindly provided us with additional and essential documents and articles.[3]

A second knowledge-base is concerned with the Etiologies of Hypertension[4] (EH).

3.2. Plan base

The second part of the non-alterable knowledge source in this system is the plan base, designed to allow the system to order its requests for new data in a coherent way. The plan base is organized as pre-ordered lists or classes of signs. Classes correspond to a classification of signs according to a method for clinical approach considered as basic knowledge. Signs of the first class, for instance, are more easily available than others, or concern the patient's own medical history and should thus be asked first. Thus, the plan is a directing mechanism which will influence the focusing of the system on the various hypotheses during the problem solving. Shifting from a sign relevant to one hypothesis to one relevant to another does express, in this system, that the point is to evaluate a cluster of hypotheses, each as probable as the others.

3.3. Results

The BCPA knowledge-base comprises about 50 rules. It has 60 signs to deal with, and 9 possible final hypotheses. The EH base has 45 rules and 8 goals.

- Following is an example of rule, expressing a *constraining link* and pointing to a *subgoal*:

---

[2] Dr. Nicole Zygelmann-Athea, Department of Reproductive Medicine and Endocrinology, Hopital Necker, Paris, was the expert consultant.

[3] Dr. Regine Sitruk-Ware, Department of Reproductive Medicine and Endocrinology, Hopital Necker, Paris.

[4] Established in collaboration with Dr. R. Nahmias, Department of Pediatrics, Hopital Necker-Enfants Malades, Paris, and Dr. R. McDonald, Department of Clinical Pharmacology, Presbyterian Hospital, Pittsburgh.

```
(defrule
 (possible-current-pregnancy nulla-gesta history-infection-uterus-or-anexes
  current-genital-infection anti-coagulant-treatment
  hemorragic-disease
  valvular-heart-disease)
 (and
  (null nulla-gesta)
  (no possible-current-pregnancy)
  (no history-infection-uterus-or-anexes)
  (no current-genital-infection)
  (no hemorragic-disease)
  (no anti-coagulant-treatment)
  (no valvular-heart-disease))
 IUD-ALLOWED ())
```

- Following is an example of a *suggestive link* pointing to a *goal,* in the context of High-blood-pressure:

```
(defrule
  (systolic-high-blood-pressure)
  (yes systolic-high-blood-pressure)
  HYPERTHYROIDISM ())
```

We have also tested the system when both knowledge-bases are loaded, thus using a larger base of nearly 100 rules. Results were satisfactory, even though the two fields are different. As expected (though being aware that the test was quite peculiar), when entering a (female) patient with hypertension, and if the latter were taking the pill, the program would open to evaluation its knowledge about birth-control, and evaluate the patient's status with regard to this problem, eventually proposing both diagnostic hypotheses for the hypertension's origin and advice as to which birth control method to switch to. Should the two knowledge-bases have no sign in common, no interaction could happen. Here, considering the algorithm's method of differential diagnosis, the *gate* between the two bases is the sign "pill". Such developments imply a common *dictionary* for the various domains.

4. Problem-solving module

In this section we present a rule-matching algorithm which can be directly used for teaching or consulting purposes. It is not meant to provide the user with a precise diagnosis but rather a cluster of the few most likely hypotheses and why they were selected, thus structuring the problem. This is done in a very specific area, clearly defined by the knowledge base. The physician could possess many such small modular knowledge bases, easily modify them and perform a problem-solving task in *a particular aspect* of the problem. Such modular knowledge bases are very easy to handle and to *build,* and are a great advantage in interactions with experts, as we experienced. Moreover, as mentioned before, bases can be linked, allowing the system to focus on various domains at a time. Presently, we are primarily interested in this algorithm as a tool modeling the necessary problem-solving module of the general system.

4.1. Method

The purpose of this first algorithm is to yield an *internal representation of the problem*. NCLOSE was originally designed for hypothesis evaluation in medicine [37]. Given a configuration of the external reality, it structures its internal knowledge into the *best-match* arrangement to that reality. This description of the problem is used to determine a *solution*. We present the main considerations in medical problem solving that led to the design of this algorithm. Its functions and features were all derived from such reflections. We will then give a formal description of the control-structure we designed, and of the additional mechanisms which enhance its efficiency and allow its use within the general learning system.

4.1.1. Evaluating hypotheses in Medicine

This algorithm is derived from a formalization of some aspects of medical problem-solving, and in particular, of the task of evaluating a set of initial hypotheses by performing a *differential diagnosis operation*. The aim of medical problem-solving is not considered here to be solely the formulation of a diagnosis, but to reach an understanding of a fundamentally ill-structured problem [35,43] by limiting and structuring the problem's space. After the initial hypotheses have been generated, the evaluation process is initiated. A number of aspects must then be taken into account [16]:

Two diagnoses are said to be *differential* if they share a common reason for being evoked. Two usual heuristics are (i) inquiring about the symptoms shared by the diagnoses, (ii) inquiring about the discriminant symptoms. However, the task of analysing differential diagnoses is a fundamental general heuristic in diagnostic performance [35].

Ordering of the rules is a major issue. When the initial set of hypotheses is assessed, we assume that the probability of each is basically equal If one clearly stands out of the group, then the others should not appear at all in the latter. In essence, the initial set has no order. We can say that the physician performs a multi-hypothesis, global approach in the evaluation of the first set of hypotheses, by first collecting easily available data. Furthermore, patient approach often follows quite definite protocols where types of questions have been determined and ordered. This is represented by the use of a plan base.

Data gathering:eithor limited to the system's will or to the expert's, is not representative of the doctor-patient interaction. Collection of data must be program-driven as well as patient-driven by means of a permanently available volunteering mechanism.

Important data are those which confirm or reject an hypothesis. All findings actually follow this rule. Nevertheless, important findings might be those put forward as such by experience. Thus we adopt no weighting method, but experience must be taken into account. Moreover, data which concern the same hypothesis are ordered similarly as for the set of hypotheses, according to the Plan base.

Physicians must constantly face uncertainty and deal with *unknown parameters*. When an item of information is unknown, it is stored in a specific memory. This memory has no term, but is actually embodied in the present state of the physician's mind. Thus, such data are constantly remembered as unknown and must be available for immediate updating and quick evaluation of the effects of belated information.

Depth of search and *focus of attention* is handled in a very optimized manner by the physician. Instead of pursuing a goal at some risk or cost, powerful heuristics allow physicians to come back to another higher level of investigation if there is there any data yet to be collected. It is assumed, then, that the physician's task is performed at various levels, sequentially, and that jumping to a deeper level, or plane, is only done when the problem has been as efficiently structured as possible on the upper plane. We will be concerned here with the structuring and limiting task on one level.

While the scope of the project embodies the problem of generation of initial hypotheses, this algorithm performs the *evaluation* of hypotheses, even though it generates new hypotheses during this process. How new hypotheses are generated is simply dependent upon the patient, not upon the system's experience, as the latter does not modify the control structure. The preceding points must be clearly formalized in order to build a tool suitable for use in a larger experiment, and whose mechanisms are to be fully traceable. These points will be discussed later however, and a more general and powerful model for a medical expert system involving various depths of investigations and knowledge will be suggested.

4.1.2. Computational aspects

The formalism adopted for this module is that of Production Systems [47]. Knowledge is represented by rules which are made of a conditional left-hand side (LHS), a list of relevant signs acting as a context, and an active right-hand side (RHS) modifying attributes in the working memory composed, in turn, of the object and goal memories. The control structure, or recognize-act-cycle, is based on the principle of *differential diagnosis* as defined in medical problem-solving tasks. It involves cycles of backward/forward searches in the graph of rules, followed by sequential evaluation of triggered rules. Conflicts are resolved using a plan-base allowing classification of signs and rules. Firing of triggered rules affects the object and goal memories. After each cycle of evaluations, modifications of the goal memory are recorded and a fixed point test of comparison to the previous state is performed. Further modifications imply further propagation and evaluation in order to complete coverage of the problem's space. Final outputs are formatted for further processing by the KAA module.

The various elements, at all levels, are lists. For clarity, we adopt a notation for the following subsection:

- H for the set of hypotheses of the goal memory
- *Hinitial* ?or ^e set of initial hypotheses
- *Hfinal* for ^e sct: of fi^na^ hypotheses
- *hi* for hypotheses or goals
- R for a set of rules
- r_tj for the jth rule of the ith hypothesis
- /y for the list of relevant signs of r_{ij}
- S for a set of signs
- $^{s}ij_tk$ for the kth sign in l_tj

4.1.3. Object representation

The patient is represented by a list of signs or terms in the object memory associated with values dependent upon him or her. This representation is similar to that found in systems such as MYCIN[42], INTERNIST [27,34] or OPS5 [18,19].

4.1.4. Initial input

The input to this algorithm is actually the list $H_{ini(ia}[_t$ The choice of $H_{initia}i$ is considered in section 6.2 with the hypotheses generation problem.

performed at various levels, sequentially, and that jumping to a deeper level, or plane, is only done when the problem has been as efficiently structured as possible on the upper plane. We will be concerned here with the structuring and limiting task on one level.

While the scope of the project embodies the problem of generation of initial hypotheses, this algorithm performs the *evaluation* of hypotheses, even though it generates new hypotheses during this process. How new hypotheses arc generated is simply dependent upon the patient, not upon the system's experience, as the latter does not modify the control structure. The preceding points must be clearly formalized in order to build a tool suitable for use in a larger experiment, and whose mechanisms are to be fully traceable. These points will be discussed later however, and a more general and powerful model for a medical expert system involving various depths of investigations and knowledge will be suggested.

4.1.2. Computational aspects

The formalism adopted for this module is that of Production Systems [47]. Knowledge is represented by rules which are made of a conditional left-hand side (LHS), a list of relevant signs acting as a context, and an active right-hand side (RHS) modifying attributes in the working memory composed, in turn, of the object and goal memories. The control structure, or recognize-act-cycle, is based on the principle of *differential diagnosis* as defined in medical problem-solving tasks. It involves cycles of backward/forward searches in the graph of rules, followed by sequential evaluation of triggered rules. Conflicts are resolved using a plan-base allowing classification of signs and rules. Firing of triggered rules affects the object and goal memories. After each cycle of evaluations, modifications of the goal memory are recorded and a fixed point test of comparison to the previous state is performed. Further modifications imply further propagation and evaluation in order to complete coverage of the problem's space. Final outputs are formatted for further processing by the KAA module.

The various elements, at all levels, are lists. For clarity, we adopt a notation for the following subsection:

H for the set of hypotheses of the goal memory
$H_{initial}$ for the set of initial hypotheses
H_{final} for the set of final hypotheses
h_i for hypotheses or goals
R for a set of rules
r_{ij} for the jth rule of the ith hypothesis
l_{ij} for the list of relevant signs of r_{ij}
S for a set of signs
$s_{i,j,k}$ for the kth sign in l_{ij}

4.1.3. Object representation

The patient is represented by a list of signs or terms in the object memory associated with values dependent upon him or her. This representation is similar to that found in systems such as MYCIN[42], INTERNIST [27,34] or OPS5 [18,19].

4.1.4. Initial input

The input to this algorithm is actually the list $H_{initial}$. The choice of $H_{initial}$ is considered in section 6.2 with the hypotheses generation problem.

4.1.5. Control Structure

Starting from an initial set of hypotheses to evaluate, the algorithm will function in a two step process resulting in an updating of the working memory on patient data and hypotheses or goals. At the same time, it is building a representation of its progress in the form of a list containing the trace of the session. The two steps are as follows, and constitute one *cycle* of performance: *propagation* and *evaluation.* These two steps represent the system's *recognize-act-cycle* or *RAC*. They are a constant in the program's approach, and are not to be modified by the learning process. Evaluation of the triggered rules affects the working memory and particularly the list $H_{considered,i}$ of hypotheses the program has confirmed to *any extent.* After each cycle, $H_{considered,i+1}$ is compared to $H_{considered,i}$ in a *fixed-point test* (see figure4-1).

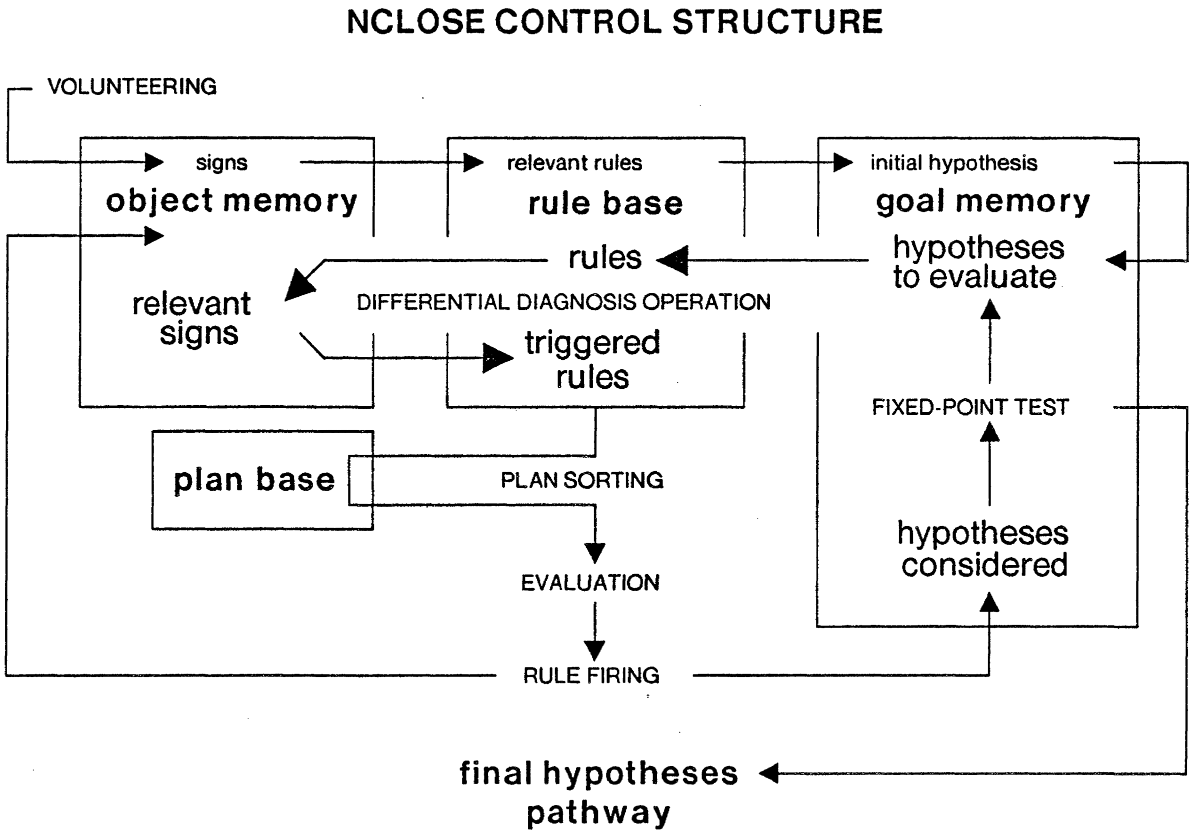

Figure 4-1: General mechanism of the NCLOSE algorithm

4.1.5.1 Propagation

Propagation is the *expansion* phase of the RAC. From the limited number of initial hypotheses, the differential diagnoses will be reached without any constraints. Thus, the problem's space is extended to *possible alternatives* to the initial formulation even before the latter is evaluated. In effect, the evaluation of a set of rules confined to the initial set of hypotheses will not take place independently of the possible other diagnoses. More specifically:

- The list of initial hypotheses $H_{initial}$ to be considered is *exploded* into the list of rules pointing at them, called $R_{initial}$. The latter is further exploded into a list of all their relevant signs called ($S_{initial}$). This process is independent of any evaluation of the rules' LHS, and uses only the *union* of the lists $l_{i,j}$ from $R_{initial}$. This mechanism corresponds to operating a *backward chaining* progression into the graph of rules, from the goal level back to the object level.

- At this point, the task *switches* to a *forward chaining* process. Let us denote by $r_{triggered,i,j}$ the *triggered* $r_{i,j}$. The list $R_{triggered}$ of $r_{triggered,i,j}$ containing at least one element of $S_{initial}$ in their $l_{i,j}$ is established, whatever goal the $r_{triggered,i,j}$ point to.

By means of the rules' RHS, $R_{triggered}$ points to $H_{triggered}$ which is a *differentially extended* version of $H_{initial}$. Thus, the scope of attention has *increased*, and the problem space also. Let us now denote $H_{initial}$ by $H_{considered,0}$.

4.1.5.2 Plan interaction

A pointer is maintained to each $r_{triggered,i,j}$, which allows its classification. The pointer is derived from exploding the $l_{i,j}$, ordering its elements, and representing this order by a word. The rules can then be ordered alphabetically.

4.1.5.3 Rule evaluation

The *evaluation* phase is the *constraining* phase of the RAC, as at this point the delimitation of the problem space becomes patient-dependent. *All* the $r_{triggered,i}$ are evaluated. There is no particular procedure for *conflict resolution.* Conflicts are actually handled by the plan and in the structure of the knowledge base.

4.1.5.4 Optimization procedures

Before any triggered rule is evaluated, its LHS is scanned in search of signs which have already been allocated a value that does not allow instantiation of the rule. In this condition, the rule is discarded, and the otherwise necessary data for its complete instantiation will not be asked. When a subgoal is scanned, the procedure analyses its premises in a recursive manner until there are no subgoals left, unless a premise is discarded before.

Plan-based classification is used to order the signs present in the LHS which have no value in the object memory. When an unknown value is encountered, further processing is stopped until the information is provided. Once provided, the particular condition in the LHS relating to this sign is tested before gathering new data for the same rule, and other $r_{triggered,i,j}$ are also rescanned for optimization.

An additional feature can be added which accounts for the system's handling of failures to confirm previously generated initial hypotheses.

- At the end of each session, H_{final} is compared to $H_{initial}$. For each $h_{initial,i}$ not belonging to H_{final}, the system knows which signs are responsible for its rejection.

- To these signs is then associated a *switch-pointer* indicating that it did reject an original hypothesis.

During the next session, when a sign with such a pointer is evoked, the forward propagation which allows it to select the very initial hypotheses is affected as follows:

- Rules are triggered, provided that at least one of the signs belongs to their list of relevant signs.

- The triggered rules with at least one sign *with a switch pointer* are *evaluated before* their goal is inserted in the list of hypotheses.

Pointers are currently irreversible. However, a semantic error in differential diagnosis operation might occur, for the propagation might never reach a given area containing a problem relevant to a different sign. In order to prevent this, the initial control is modified as follows:

- When dealing with a pointed sign, backward/forward propagation is performed *before* the evaluation of the rule, as for ordinary rules. Thus the goal of the rule might not be retained in the initial hypothesis, but initial propagation will occur *as if it had been retained.*

Therefore, we have a *failure-driven* optimization which adds a constraint at the level of the generation of initial hypotheses. However, further propagation is not affected by these pointers.

4.1.5.5 Unknown data

When some data is unknown, the rule it is concerned with remains in the $R_{triggered}$ although it cannot be fired. The volunteering facility allows the user to introduce any new data at any time during the session, particularly previously unknown data. In the latter case, concerned $r_{triggered,i,j}$ can be then evaluated.

4.1.5.6 Rule firing and memory update

A given rule may only be fired if all its predicates are verified. The strategy adopted for rule firing is *irrevocable* [30]; hence a rule cannot be fired twice. Two kinds of memory update may result:

- Modifications in the object memory of signs or subgoals, usually avoiding unnecessary data gathering in the same problem context.

- Modifications in the goal memory that give to a verified hypothesis a value represented by the list of *conditions* of the relevant instantiated rule. This value is thus self-explanatory.

4.1.5.7 Closure function and fixed-point test

When all elements of $R_{triggered}$ have been evaluated, the goal memory has undergone all possible modifications. Hence, the program makes the set or list $H_{considered,i+1}$ of $h_{triggered,i}$ which now *have* a value attributed during this cycle or previous ones. This set might contain the complete or part only of $H_{initial}$ plus other $h_{considered,i}$. Thus, the whole operation is a *closure function.*

The *fixed-point test* performs a comparison between the initial set at the beginning of the cycle and the resulting set at its end.

- If the two sets are *different*, new hypotheses have been confirmed by rules selected by the *differential propagation.* The new hypotheses that have been evoked and generated or confirmed to a certain extent must now be *fully evaluated*, and a new cycle will be initiated.

- If the two sets are *similar*, no new hypotheses are to be considered and the output can be proposed. In the first cycle a hypothesis might not be confirmed and no differential one generated; thus if the resulting set is *smaller* than the initial one, the fixed point test is also positive.

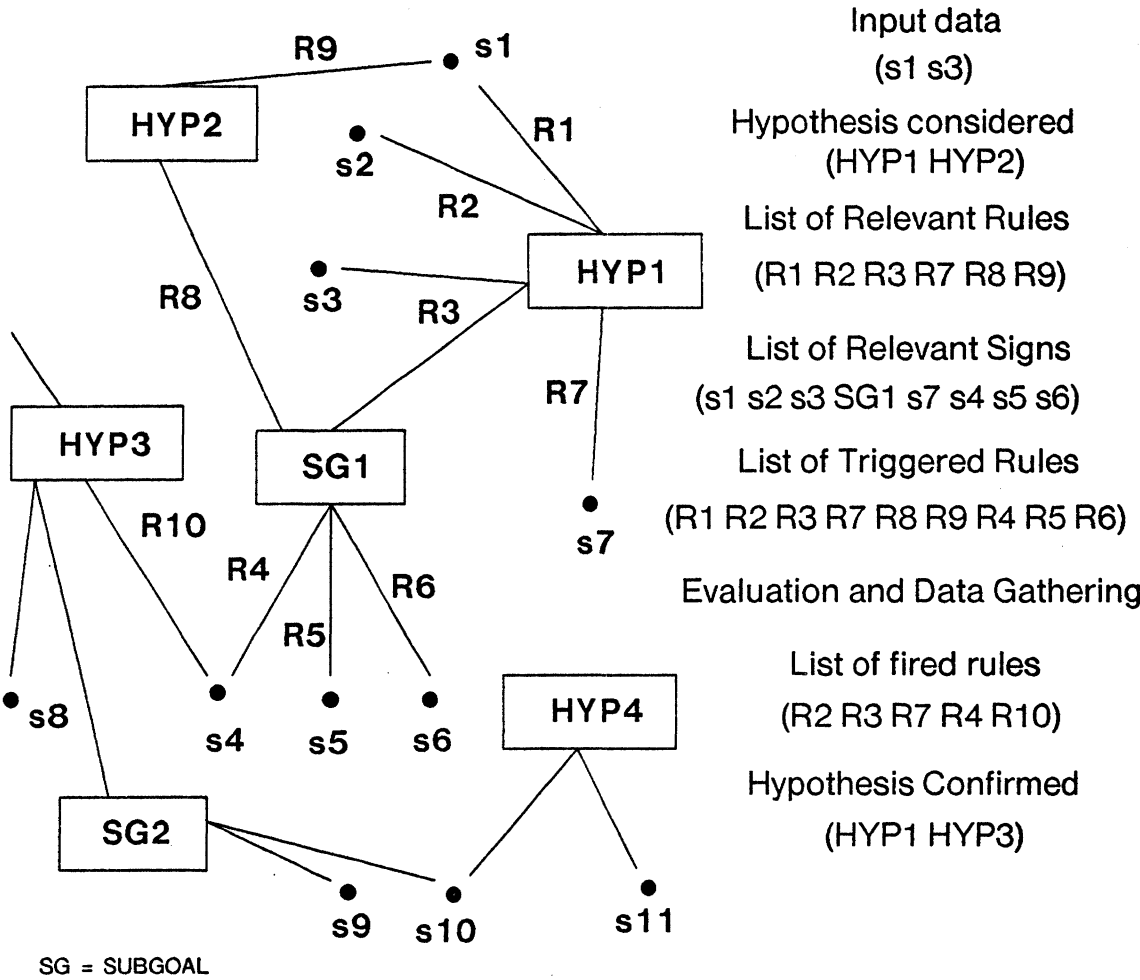

Figure 4-2: Simple model of propagation and control

4.1.5.8 Output representation

Once the problem space has been delimited, the structure of the problem itself can be found in the *trace* of the control mechanism. The information readily available at the end of a session can be summarized as follows:

1. S_{total} with values of its elements in the object memory

2. $H_{initial}$

3. H_{final} with the respective values of its $h_{final,i}$ in the goal memory

4. $H_{rejected}$

5. $R_{triggered}$ decomposable into R_{fired} and $R_{discarded}$

The main output of the system is the final set of hypotheses. However, the complete information about the structuring task of the program allows various types of output information, in particular *solution paths* [44], to be transferred into the next module, which builds the dynamic-memory. The output has the form of a list which for instance might contain ($H_{initial}$, R_{fired}). Moreover, as lists or sets are updated during the process, the order of events is conserved within them. We will refer to them as *pathways.*

In the present case, we have studied *three kinds* of pathways, which are actually lists of *signs*, representing three different kinds of trace.

- $S_{positive}$ of signs $s_{positive,i,j,k}$ responsible for the *success* and firing of the rules $r_{fired,i,j}$ of R_{fired}. When building the dynamic memory using these paths, the system should infer an expectation composed of the most representative of the *expected* signs.

- $S_{negative}$ of signs $s_{negative,i,j,k}$ responsible for the failure of the rules $r_{discarded,i,j}$ of $R_{discarded}$. In this case, the expectation will be composed of the most representative of the *uncommon* and thus *unexpected* signs.

- $S_{negative and positive}$ represented by the *intersection* of $S_{positive}$ and $S_{negative}$. These signs are the most discriminant ones. However, some signs might never verify such a condition in a given knowledge base. Therefore the structure of the rules interferes with this criteria.

Thus, we can obtain an *internal representation* of the problem which actually yields a new knowledge representation where new external information is encoded in the combination and ordering of elements of the fundamental rule-based knowledge.

4.2. Results: example of a session

The NCLOSE module was tested on actual case records, once the knowledge base could cover the domain adequately. Testing was done following a simple protocol: (i) patient's complaints are volunteered, (ii) the program is run and asks for further informations, generally all available, (iii) the output is compared to the the actual attitude of the expert in the case.

Computing time is very short, particularly with the ZetaLisp implementation on Lisp Machines where time for the user to answer is the limiting step. Results showed an excellent success rate in reproducing the expert's opinions and actions. Results depend upon the expert's point of view, and another one might not rate the system as well.

An example of a session in birth-control advice is presented in Appendix I.

5. Knowledge Aggregation Algorithm

5.1. Introduction

The performances of the NCLOSE subsystem are dynamically stored to build a higher level knowledge. The latter, called *concept knowledge* in the following sections, is abstracted from successive evaluations of real cases.

The KSI subsystem makes use of this concept knowledge to find *first-look* signs. In designing the KAA module we tackle two different but related issues:

- Unsupervised Learning: Clustering techniques are used to aggregate clusters of pathways [15].

- Expectation-failure control structure: Using minima of a given criterion as expectations and distance considerations, we evaluate the difference between expected and actual input.

The underlying assumption is that the inference of concept knowledge is the result of dynamic alterations of internal structures that represent the environment. The content and organization of these structures are precisely this higher level knowledge which is the "expertise" necessary to *first look.*

5.2. Method

5.2.1. Reasoning Pathways representation

The inputs to the subsystem are *reasoning pathways* obtained by the NCLOSE subsystem, performing differential diagnosis on a medical case. They are represented as the list of rules triggered during evaluation of the case, along with their LHS, RHS, goal and relevant signs. Initial hypotheses and final confirmed goals are also present. Thus if P_i is such a pathway :

$$P_i=\{R_{i,j}\}$$

where $R_{i,j}$ are the rules fired during the evaluation.

Moreover, a set of signs built up from the signs present in the relevant lists of the $R_{i,j}$ is associated with the path P_i. Three distinct methods of association were studied and tested:

- The associated set of signs is the union of the relevant lists of the rules pertaining to the path. These are precisely the signs involved in rules which confirmed one or several hypotheses (i.e. fired rules).

- The associated set of signs consists of the signs involved in rules which were triggered but not fired.

- The associated set of signs consists of those signs involved both in one or several fired rules and in one or several triggered but not fired rules. This set is the intersection of the two preceding sets.

Thus, pathways are actually stored as a set of signs, according to one of the previous methods of association.

5.2.2. Symbolic distance

Associations of relevant signs or related rules are entities a physician is likely to consider when reasoning. We designed a *symbolic distance* between two reasoning pathways, based on the analysis of such entities. The distance, or *proximity*, of two reasoning pathways is a set of signs resulting from the comparison between the two pathways. This proximity, the symmetric difference, retains the signs that make the pathways different from each other, and thus has a specificity flavor. (See Appendix for a mathematical definition of this proximity.) As pathways enter the aggregation module, the different proximities between pairs of pathways are stored in a dynamically updated matrix: the *similarity matrix.*

5.2.3. Symbolic Concept Aggregation

The purpose of the aggregation algorithm is to compute clusters of *relevant* pathways. For each proximity present in the similarity matrix, a partition of the set of pathways is computed. Let $Part_i$, $C_{k,i}$ and E denote respectively the ith partition, the kth cluster of this partition and the set of all pathways; then the result of the clustering analysis is defined by: n sets of signs, the distinct proximities of the similarity matrix, t_1 to t_n, and $\forall\, i$ from 1 to n, $E=\cup_{k=1}^{p} C_{k,i}$;$C_{k,i} \in Part_i$ with $C_{k,i}=\{P_{j,k,i}\}$ and p the number of clusters in this partition.

Inside the partition $Part_i$ indexed by t_i the following property holds:

- For $C_{k,i} \in Part_k$, $d(P_{p,k,i}; P_{q,k,i}) < t_k$ where d stands for a measure of proximity and $<$ for inclusion.

Clustering analysis [15] provides algorithms allowing to compute such partitions from a distance matrix. The process is known as the transitive closure of a relation and is described in Appendix II. For the current description, we only need to know that this process involves the computation of a similarity matrix from the original distance matrix before actually building clusters. However the model of symbolic distance used was such as to yield directly a similarity relation between the different pathways, as demonstrated in Appendix II. Hence the transitive closure process was reduced to the sole partitions building phase.

Given the proximity of the similarity matrix, a partition of the set E of all pathways results from the following incremental process:

- Step 1: A path is chosen among the pathways which are not already pertaining to a cluster (if no cluster exists, the path is selected at random in E). If no remaining path exists, the partition is constituted of the current set of clusters.

- Step 2: This path is a *seed* for a new cluster. Among the pathways which are not already pertaining to a cluster, those pathways with a proximity to the the seed contained in the given proximity are joined to the seed in the currently built cluster. The process jumps back to step 1.

This operation is iterated for the distinct proximities of the similarity matrix.

The distinct elements of the similarity matrix are indexes to partitions. Furthermore, inside a given cluster of a partition, the proximity between two pathways is contained in the index of the partition. The global structure thus defined is called *Concept Knowledge Network.*

5.2.4. The Concept Knowledge Network

The concept knowledge network contains several partitions of the set of reasoning pathways, each of these partitions being indexed by a list of signs resulting from the whole process of clustering described above in this section. Each cluster of pathways constituting a partition is representative of a *concept.*

Hence,

- A partition $Part_k$ is associated to each distinct element t_k of the final similarity matrix.

- Each partition $Part_k$ is a set of p mutually exclusive clusters $C_{k,l}$ such that $E = \cup_{l=1}^{p} C_{k,l}$.

The growth of the Concept Knowledge Network is *event-driven* in the sense that the network is dynamically updated and altered as the reasoning pathways are memorized. Each partition is considered as a level of abstraction containing concepts represented by clusters. In this framework, concepts appear as sets of *related* or close reasoning pathways used by the problem-solving module. Hence a partition is a list of mutually exclusive concepts which were actually used in evaluating a real case. The diversity of partitions accounts for the experience abstracted by the system from its past performances.

The KAA module, through its clustering operation, builds a *structure* relevant to the previous history of the system on the a priori ill-structured search space, enabling the KSI system to perform an easier search for the

5.3. Results

5.3.1. Processing of the reasoning pathways

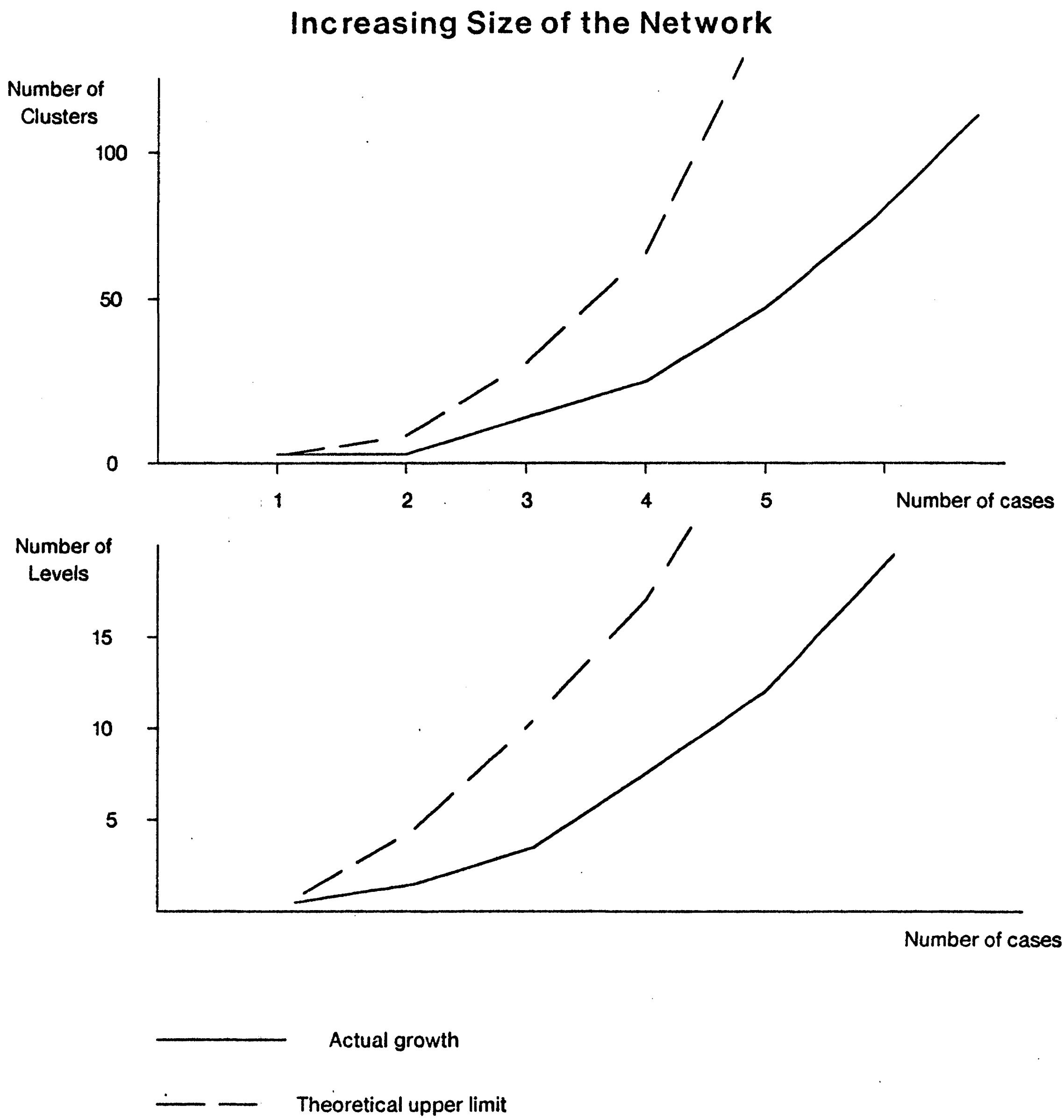

Figure 5-1: Actual vs Polynomial growth of network size

Figure 5-1 depicts the increase in two size parameters on the concept network during a recorded session: the total number of clusters and the number of partitions in the network.

The results show that the actual number of clusters and partitions are very inferior to their upper limits, respectively n^3 and n^2. This *slow* polynomial growth of the network is due to the high consistency of the Rule

Base.

The repartition of the clusters inside a partition is altered as the cases are evaluated. The average number of clusters in a partition linearly increases with the number of cases recorded. The number of signs present in each set of these clusters ranges roughly from 5 to 20, according to the cases evaluated.

5.3.2. Global behavior of the symbolic aggregation algorithm

In this subsection, we point out some global aspects of the incremental acquisition performed by KAA which are very similar to the results we obtained with our initial application of the algorithm [9, 10].

Focus of Attention: As the acquisition process goes on, the system draws expectations that reflect an increasing focus of attention. During a session where a lot of analog reasoning pathways are used, the algorithm will infer more and more specific concepts.

Extent of Alterations: As the list of input pathways grows the extent of the alterations of both structure and content of the network decreases. This is an *asymptotic behavior* of the system near a stable equilibrium, which is further reinforced if expectations drawn by the system are confirmed, i. e. if the inputs are not very different from what the system expected. The system relies on prior knowledge.

Network alterations: The structure of the network, i.e. the set of partitions of the pathways set, is very sensitive to early expectation failures. With a small amount of knowledge, the network is fragile and subject to drastic modifications. This fragility decreases quickly as the system acquires new experience.

Growing size of the network: The size of the matrix is $O(n^2)$, and the number of partitions found in the network is also $O(n^2)$. This might be a serious drawback to the method chosen, since in each partition, the number of clusters is at most $O(n)$, and the *total* number of clusters or concepts inferred is $O(n^3)$ which is unrealistic as n increases.

Multiple inclusions: The counteracting effect is that there exist either multiple occurrences of the *same* cluster or multiple inclusions of clusters inside others in different partitions. Actually it appears that the current number of clusters is less than the upper n^3 limit.

Partial ordering: Since there is no numerical index to sort clusters, it may happen that proximities between input and distinct clusters can not be comparable. In this case the first-look signs have to be drawn from a set of clusters rather than from a particular cluster. The interesting interpretation of this result is that from its current knowledge the system is able to suggest several clusters or concepts as expectations of forthcoming input. These distinct expectations account for distinct representations of its past acquired knowledge in reference to the input.

6. KSI: Knowledge Structure Interface

Once the dynamic memory is built and updated by the previous module, a new method is necessary in order to *interpret* the clusters of pathways and *apply* this interpretation so as to modify the system's behavior. Thus, this interface comprises two main aspects:

- The resulting network of clusters from KAA is processed and yields a list of *first-look* signs. They are representative of the system's general expectation.

- First-look signs affect the system's approach to the new patient, by suggesting data to gather in order to select a particular cluster of initial hypotheses. This interaction is the junction between the memory and the *sensory* structure.

6.1. Method

6.1.1. Symbolic determination of first-look signs

Clusters of pathways, as computed by KAA, can be decomposed into their component signs. The search is thus performed in two steps :

1. Find a restricted set of clusters *representative* of the current concept expectation of the system's dynamic memory. We call this set the set of *relevant clusters.*

2. From the signs present in these relevant clusters, compute the first-look signs.

6.1.2. Determination of relevant clusters

The determination of relevant clusters addresses the problem of search versus knowledge [28, 14, 2]. At each step of this search, some information about the goal is used to guide further processing. This information is formalized as criteria allowing rejection of subsets of clusters without further evaluation. The sequence of criteria is as follows :

- The set of proximities $p_{i,j}$ of the final similarity matrix is scanned for minimal elements with respect to the inclusion. If $p_{i,j}$ is included in $p_{k,l}$, then the first proximity is kept and the second one discarded. This is a *specificity* criterion.

- Thus the initial set of the search is the set of clusters belonging to the partitions indexed by the preceding $p_{i,j}$. Let us denote by S this initial set : $S=\{C_k\}_{k=1}^{N}$. From this initial set are kept only the maximal clusters with respect to the inclusion. This is a criterion of *abstraction*, taking into account the meaningful clusters which are aggregates of several list of signs.

- Let us denote by S_1 the preceding restricted set of clusters. For each cluster belonging to S_1 its median fuzzy set is computed, and distance between this expectation and the incoming input is minimized over S_1, thus yielding a new set S_2 of clusters that are minimal (with respect to the inclusion) and nearest to the input pathway. As two sets might not be comparable by intersection, the set S_2 is not necessarily a singleton. Furthermore we are assured that S_2 is not the empty set $\emptyset$. a *nearest proximity* criterion has been applied.

This final set S_2 of clusters is precisely the set of relevant clusters used for the determination of the first-look signs.

6.1.3. Determination of first-look signs

We are therefore left with a very restricted set of relevant clusters to compute the first-look signs from. Usually two to five clusters are present at this stage of this search.

In order to take advantage of the information encoded in a given cluster, we need a criterion pointing out the differences between the constitutive lists of signs. Since they are part of the same cluster, these lists are very similar, but they differ from each other by certain specific signs representative of specific features of the cases from which they were derived. By selecting those signs, obtained in the symetric difference of all the lists of signs of a given cluster, specific aspects of a general concept are highlighted.

Eventually, we select from this list of possible first-looks the one containing the largest number of signs leading to the minimum number of initial hypotheses. Thus, the system must generate a most precise set of hypotheses while taking into account the widest possible range of elements from its experience.

During this second step the following elements are used as guidelines :

- Specificity : we are always looking for salient and striking features of a most abstract concept (according to its position in the network, and the level of the partition chosen).
- Focus of attention and accuracy : the selection of first-look signs among these possible specific features allows the discrimination of an efficient restricted scope for the initial hypotheses without loss of precision.

6.2. Altering hypothesis generation

From previous signs, a list of rules containing one or more of them is established. From these rules, a list of initial hypotheses is built If the sign does not yet have a pointer assigned, first-look signs have their propagation switch temporarily on. The relevant rules will be *evaluated* for *the condition* in the LHS *concerned with the sign.*

In order to gather information correctly, first-look signs are ordered according to the plan base, as for any other part of a session. Optimization procedures are also used for first-look signs which might belong to the same rule without separate occurrences elsewhere in the rule base. Thus information gathering for those signs follows the same coherence as for other signs. The optimization procedures take into account the presence of subgoals. When physicians jump from the level of those signs to the level of hypotheses, we hypothesize that parallel processing is performed which allows a very rapid and accurate definition of the goals. Implementation on Lisp Machines can simulate this highly efficient computing method. This is the most suitable part of the whole system for parallel processing, for the propagation by differential diagnoses cannot follow such a course.

Therefore, a set of initial hypotheses is defined *before* collecting data from the patient. The nature of the first-look signs is *patient-independent* but *experience-dependent*, whereas their value is patient-dependent. If no further patient data is collected at this stage, the set *of first-look hypotheses* is used to trigger the evaluation process. However, should there be any sign available at first (e.g. complaints...), it is volunteered at the beginning and might increase the set of initial hypotheses. As always, any further information obtained during the evaluation can be volunteered.

6.3. General implications

First-look signs are not generated at the first session since the clustering process needs at least two cases to run. Thus the system defines an *intuitive, experience-based and patient-independent approach.* This approach is modified according to the cases encountered. It is aimed at allowing the system to optimize its search for the right diagnosis by considering the most pertinent factors issued from its past experience.

7. General Results and Discussion

In this section we present the general results of the behavior of the system (named SKP) when processing real cases provided by the expert and which serve as a control base. These results set the stage for a general discussion of the validity of the model, and new developments will be suggested in the last subsection.

7.1. Global Results

The results are to be considered from two standpoints, related to the two main objectives of the system. This is a learning system, acquiring knowledge in an unsupervised manner, imposing structure on an *ill-structured* domain in order to better perform a given problem solving task. On the other hand, this system presents a emulation of a physician's behavior. Qualitative and quantitative criteria allow evaluation of the approach with respect to both points of view.

7.1.1. The learning system

The immediate result, drawn from sessions involving processing of a variable number of cases (usually 10 to 30), in various orders of occurrences and on the two medical fields covered by the Rule Bases at our disposition, showed that the system is indeed able to structure the problem space and use this representation for improving its task performance.

The clusters built by the system, from successive evaluations, refer to actual medical therapies, or ways of reasoning in the medical field chosen.

```
{ (Age HBP Diabetes Cholesterolemia History-phlebitis-vasc-acc)
  (Age HBP Diabetes Cholesterolemia History-phlebitis-vasc-acc
                    History-mother-sister-genital-canacer)          }
```

Instance of a cluster related to macroprogestogenes

```
{ (Age HBP Diabetes Cholesterolemia History-phlebitis-vasc-acc)
  (Age HBP Diabetes Cholesterolemia History-phlebitis-vasc-acc
                    History-mother-sister-genital-canacer)
  (Age Obesity Diabetes Tobacco Chloasma-pregnancy
        History-of-phlebitis-vasc-acc Current-liver-disease
     Hyper-prolactinemia History-of-cholestasis Benign-Mastopathy
            History-of-toxemia-gravida-non-essential
        History-of-Breast-cancer History-of-prem-fam-vasc-acc)
  (Age Obesity Diabetes Tobacco Chloasma-pregnancy
        History-of-phlebitis-vasc-acc Current-liver-disease
     Hyper-prolactinemia History-of-cholestasis Benign-Mastopathy
            History-of-toxemia-gravida-non-essential
  Taking-Pill-normal-doses Good-tolerance-pill-normal-doses       ) }
```

Instance of a higher level cluster related to macroprogestogenes and estroprogestogenes

Figure 7-1: Examples of Clusters

This figure shows two typical examples of different level clusters in the set of partitions. They refer to the BCPA Rule Base. The top cluster appears in the lower level of the network, as a *specialization* of the next cluster which refers both to macroprogestogenes and estroprogestogenes therapies.

Figure 7-1 is an example of clusters the aggregation module builds from the traces of the preceding problemsoiver mouuie.

Though these structures are essentially very simple, other methods of elaboration on a dynamic memory could be used in the design of the aggregation module [21,4,49], e.g., i[difference handlers] [48], MOPS [39,40, 24], a set of meta-rules [25] or analogies [7], Also, in this system, the learning process is quite *independent* from the problem-solver, although they are actually integrated in a global process [5]. The problem-solver is affected only at the level of its input which is *pre-processed* by the concept-driven mechanism.

7.1.2. The expert behavior

We have adopted a simplified definition of *medical expertise* for the purpose of this research, based upon the physician's ability to *pre-structure* the problem, and thereby limit its space. Moreover, we postulated that this ability is the result of *compiling personal experience* and that it is not taught. Figure7-2 shows how the model we present might, in effect, simulate the acquisition of this behavior. The experiments were made as follows:

The *mode* of determining first-look signs is the selection of signs that *confirmed* a hypothesis, certainly the most common way of inferring those signs. Given the EDH rule base, the system is presented with two quite different *cases, A* and *B.* Once the first *first-look signs* are defined, a series of similar cases is entered, in any order, using the first-look data *only* and volunteering no other data. Hence, signs associated to a pointer *cannot* affect the selection of the initial hypotheses. The system is, thus, completely *concept-driven* for drawing initial hypotheses, and calls for the data-driven process only for evaluation. After about 10 such cases, a new *original case C* is presented a first time. *A* and *B* are then presented again a few times until *C* is for the second time. The same procedure will then apply to another *original case D.* Each time a given case is *evaluated,* the *constant coned final hypotheses* are given, to which the initial hypotheses can be compared. Figure7-2 gives an example of the evolution of a cluster of initial hypotheses, and'thus of an increase in the quality *of the first-look approach* for a given case.

* CASE A IS A HYPERTENSION INDUCED BY A FTBROMUSCULAR DISEASE OF THE RENAL ARTERY
* CASE B IS A HYPERTENSION INDUCED BY AN IMPORTANT STRESS
* CASE C IS A HYPERTENSION DUE TO A HYPERTHYROIDISM
* CASE D IS A HYPERTENSION DUE TO AN ACUTE GLOMERULONEPHRITIS

Accuracy of the generation of initial hypotheses by the evaluation *only* of first-look signs is estimated by comparing the size, and the medical relevance of the initial set of hypotheses. It is compared to the set of final hypotheses. Experiment 7-2 shows:

1. At the begining of the experiment, when two cases are presented in any order a number of times, *the first* first-look generations are not very accurate, and remain stable. Thus, we decide to present a *new* case C, quite different from both *A* and /?, with very few signs in common. However, *C* must have at least one of the first-look signs with a verified condition. Thus, the new experience must be *somehow* even at minima linked with the previous ones in this type of experiment where no other data is volunteered.

2. When patient *C* is encountered for the first time, the first-look, based on the previous experience is not efficient. However, this sole occurrence of *C* has modified the system's expectation and allows a rapid, efficient recognition of the second occurrence, at some distance, of the same case. This behavior is fundamentally *non-probabilistic,* as the many occurrences of cases *A* and *B* would

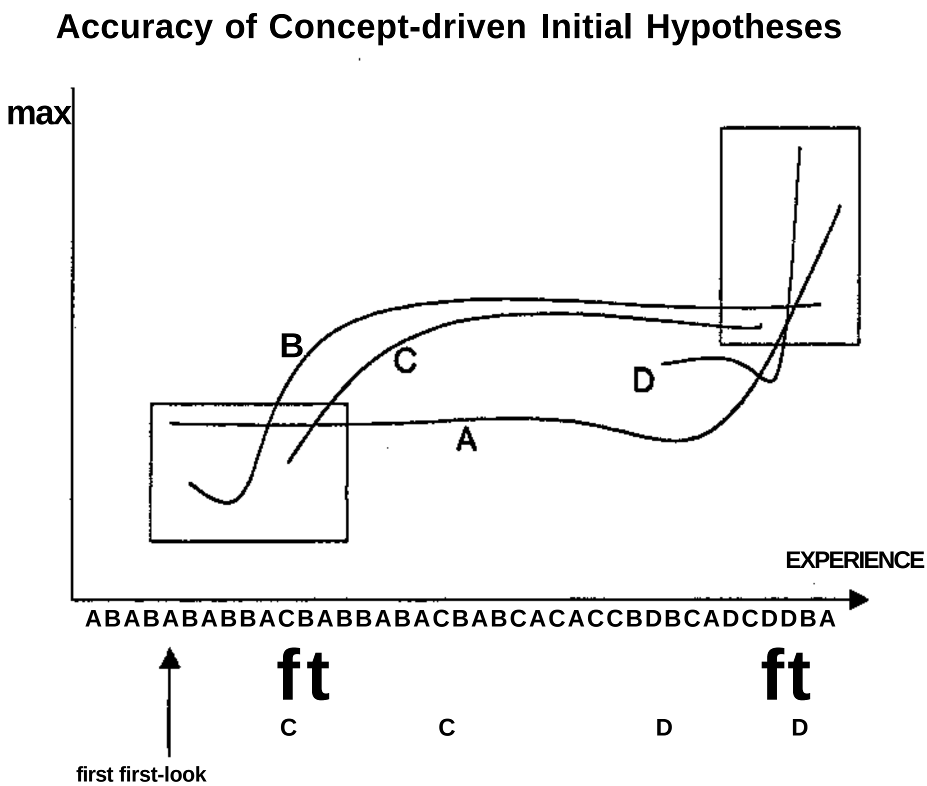

Figure 7-2: Acquisition of first-look generation of hypotheses

This figure shows in curve-fitted lines the evolution of the accuracy of the initial cluster of hypotheses yielded by the evaluation of the *first-look* signs when considering patients A, B, C and D. In abscissa, the course of the system's experience is shown, with the series of cases it encountered. Arrows point to the *important events.* The two rectangles indicate the general situation of the system with regard to its first-look generation capacity, at the beginning and further during its experience.

prevent the noticing of *C* as an *interesting* entity, which is not the case here.

3. The same phenomenon is observed with Z>, easily and efficiently recognized after two occurrences.

4. New cases *enhance* the recognition of long known ones, as is seen with *A* and *B*. The principle of differential diagnosis, as formalized here, is at the basis of this important effect. For instance, the introduction of the *C* event in the dynamic memory will somehow modify the list of first-look signs in a manner influencing the choice of initial hypotheses when confronted to *A*. Similarly, the recognition of *B* is affected by the exposure to *C* but not by the exposure to Z). This may be by adding a new sign, or by deleting or changing a sign already present in the list

Let us consider the first-look signs, and the cases they are relevant to, *before* the event C:

RAPID-ONSET (A)
SEVERE-HIGH-BLOOD-PRESSURE (A)
SYSTOLIC-HIGH-BLOOD-PRESSURE (B)
ABDOMINAL-BRUIT (A)

These signs are not modified by die perturbation itself, but only after cases *A* and *B* were presented again. After case 5, we obtain:

ANXIETY (A, B)
PALPABLE-THYROID-GLAND (C)
WEIGHT-LOSS (C)
PALPITATIONS (C)
PERMANENT-TACHYCARDIA (C)
SYSTOLIC-HIGH-BLOOD-PRESSURE (B)
ABDOMINAL-BRUIT (A)

Thus, SEVERE-HIGH-BLOOD-PRESSURE and RAPID-ONSET were removed. The new set of signs is quite representative of the three types of patients. It must be noted that it remains invariant until *D* is met. Changes then also occur, and the set is diminished, giving for instance:

SEVERE-HIGH-BLOOD-PRESSURE (A)
AGE (A,B,CD)
PROTEINURIA (D)
RECENT-STREPTOCOCCAL-INFECTION (D)
ABDOMINAL-BRUIT (A)

7.1.3. Use of first-look signs

The evolution of the system tends to reach a *general optimal* state of *expectation,* as shown by the two rectangles. However, it is conceivable and it does happen that the quality of the expectations is *lowered* It is obviously the case if no first-look data can be gathered from the patient; the initial hypotheses will depend on the volunteered data. Thus, cases that are *too* irrelevant to the previous experience might affect the behavior towards one or several previously encountered cases.

The set of first-look signs remains within reasonable sizes. The mechanisms by which they affect the initial hypotheses generation varies. A case might be better approached because a sign was *deleted* from the list that induced the selection of a wrong hypothesis, or one might be *added* which now helps discriminate better. In any case, the signs are also *evaluated and* their presence in the list is not enough by itself. Moreover, first-look sign sets may vary or oscillate according to the type of experience.

7.1.4. Medical Interpretation

The knowledge bases were made with experts, and case records are being used for testing. The results show that the problem of *hypotheses generation* [16, 29] is indeed complex, and may be approached by techniques such as those presented here. This system enhances its capacity to solve problems. In many instances, the process of evaluation that follows the generation of the hypotheses seems "useless" as the right answers are given at *first sight.* However, this is still only a *pre-structuration,* since it can only contain a small part of the problem's structure (first-look signs number is between 4 and 9 in figure 7-2).

Examination of the lists of first-look signs shows they contain condensed information on past experience, updated by the new ones. It is also the case when interpreting the. so-called *unexpected signs.* This system bears an *intrinsic instability* essential to its behavior. No experiment looks *exactly* like another due to the numerous parameters that may change. However, the behavior we outlined is highly reproducible.

Finding the right initial hypotheses does not necessarily diminish the number of questions to be asked. This would be the case if the number of hypotheses were drastically reduced, but this is not very common. The main reason is that the system will i[check] all it wants, and in particular will search for differential diagnoses. Another reason is the relatively small size of the knowledge bases and their specificity. Thus, the *posterior effects* of generating correct hypotheses from experience could not be precisely studied.

The same study as in figure7-2 was undertaken with the volunteering of patient data. In this case, and with the help of the pointers for evaluation, the evolution of the proper effects of first-look signs was clearly overshadowed. Indeed, when the patient complaints are added to the data for the first-look signs, the system becomes more precise. In the experiment described above, if the presence of an ANXIETY is volunteered when presenting the case *B*, the set of initial hypotheses becomes equal to that of final hypotheses, i.e. the program considers "stress" and "hyperthyroidism" at first. This result means the physician must consider hyperthyroidism as well, on the basis of the presence of a *systolic* high-blood-pressure as the only favorable hint.

7.2. Instability and Perturbations

In this section we present results related to the behavior of the system when processing occasional unexpected cases. First-look based on unexpected signs has been used, in order to show how an *a priori* general estimation of the unexpected signs becomes more accurate and specific when particular instances of unexpected cases are encountered. Moreover the results show how the stability of first-look signs is affected by such perturbations.

In this experiment, two populations denoted by *A* and *B* are considered. These are very distinct groups with respect to patient's symptoms. The system is presented, in a first session, with cases exclusively issued from population *B*. A perturbation is induced by presenting a case from population *A*, before continuing with more cases from population *B*. As a result first-look signs now contain only the specific unexpected signs allowing a better processing of a new case from population *A*. This is a process of specialization, drawn from actual instances of unexpected cases. Though concerned with population *B* patients, the system is able to quickly detect population *A* cases and process them correctly.

After the last occurrence of case *A*, the first-look signs reflects a strong concern with *A* cases. This first-look is again subject to evolution according to new incoming cases.

The preceding results illustrate the fact that whereas expected common cases are handled on the basis of the greater amount of recorded similar cases, unexpected cases can induce specializations allowing new similar unexpected cases to be handled on the basis of the previously encountered *particular* instances. Another interesting conclusion, it now appears, is that the system is unstable in the short term, one case being enough to drastically alter first-look signs, but remains stable in the long term as the statistical weight of population *B* overcomes the *A* perturbations.

7.3. Expectations and initial formulation

How physicians do generate adequate initial hypotheses is a most difficult part of medical problem-solving to understand [16, 32] and a general cognitive method has not yet been implemented. Relying on the data-driven mechanism alone has proved to be insufficient [33]. The system we described adds to the data-driven process an endogenous concept-driven mechanism represented by the expectations *inferred* from experience. Physicians do use experience-based expectations which are in fact *heuristics* issued from their *interpretation* of experience.

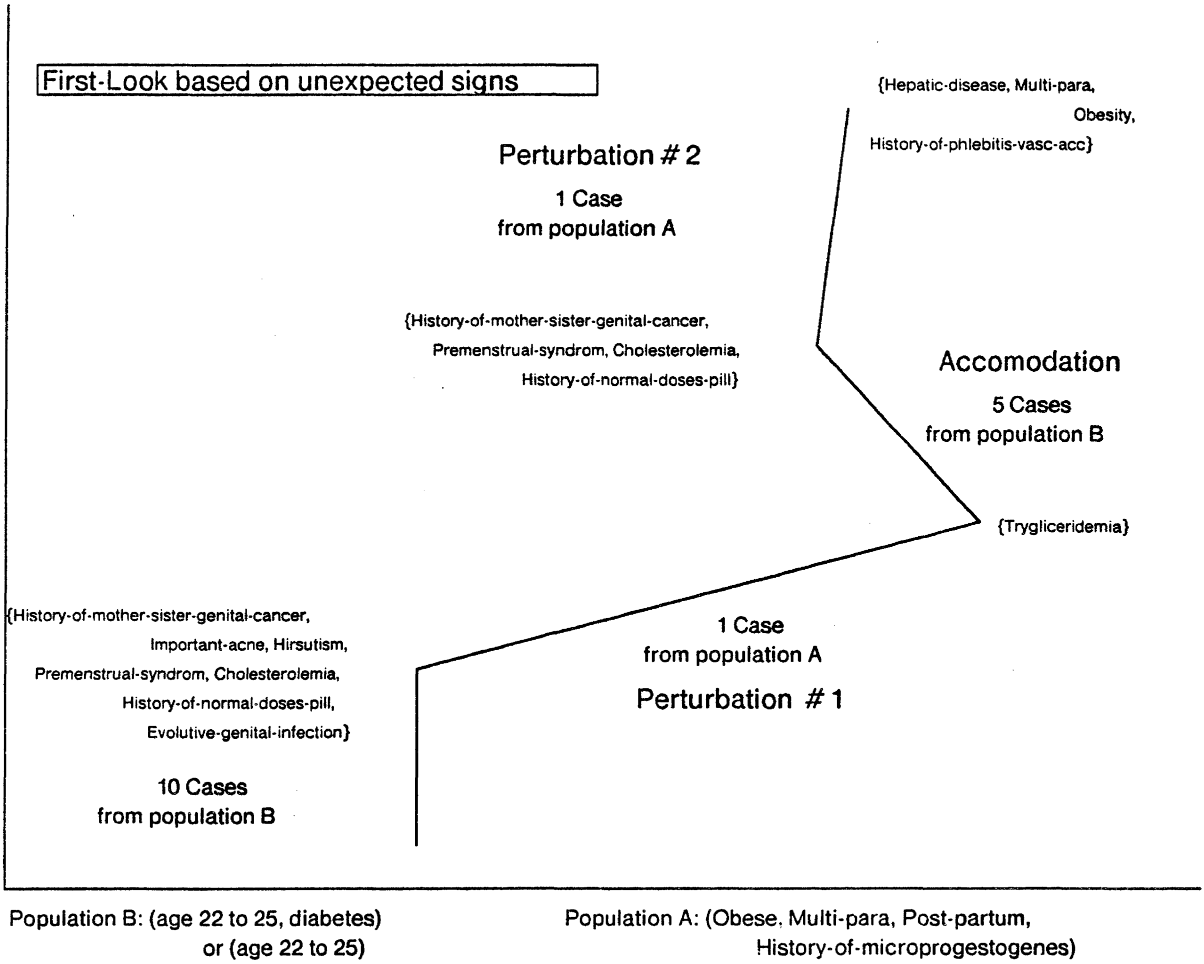

Figure 7-3: Instability and Perturbations

Sensitiveness to perturbations: Presented with ten cases issued from population *B*, the system yields a set of unusual signs, with respect to the population *B*, as a first-look. This set is completely modified by the occurrence of a population *A* case. After a period of accommodation with five more population *B* cases, the first-look allows previous correct processing of *B* cases as well as early detection of *A* cases, as is shown by presenting a new *A* case.

Certainly some of those expectations can simply be taught, but experience will reinforce them. They can be encoded directly from the expert and are thus shown to increase the system's efficiency but diagnoses are missed [1]. The latter drawback can be overcome by adopting a control structure such as NCLOSE, but the definition of a concept-driven mechanism requires the building of a *learning* system.

7.4. The alterable rule base

In the search for machine efficiency the classical approach lies in the design of an adequate rule processor [3, 36, 26, 38]. The rule base constituant of such a production system is alterable by some higher level system. Productions are allowed to be added, altered or deleted from the rule base.

Acquisition of new productions thus accounts for the fact that knowledge size or quantity increases with time. However, different investigations have emphasized that *direct* approaches are not sufficient to account for the better *qualitative* use of this knowledge shown by experts [38].

As an illustration of this idea let us consider a probabilistic or weight-oriented treatment of rules and pieces of rules. Some global features of the behavior of such a system may be pointed out: need for a repetitious style of instruction, lack of sensitiveness to perturbations, stability in the long term, easy acquisition of large amounts of knowledge, reinforced accurateness and specificity. Whereas handling common cases is easily performed since such cases are *average* cases and closely follow the system's expectations, new cases need a *special handler* as they cannot fit in the average expectation. For such a situation the flaw in the performance might be interpreted as missing information in some piece of a rule, and new rules are devised for that particular occurrence by addition, merging or other operations on the rule base. However, without the ability to draw analogies [6, 8], the system remains unable to handle related unexpected cases; because unexpected cases have, by definition, a negligible weight and a special *ad hoc* set of handling rules, specific to previously encountered particular instances of the unexpected.

7.5. An unstable system

Some peculiarities emerge from the results described in the previous section. In a system such as presented in this paper, knowledge acquisition is characterized by the following features :

Sensitiveness: whereas small perturbations remain negligible if they are still close to the general expectations of the system, striking divergences from these expectations are more likely to largely influence the forthcoming behavior of the program. This might be considered as an inherent *instability*, or more specifically one can refer to the expectation of the system as to *unstable equilibrium points*, or *relative extrema*. That does not imply however that *all* the reachable states of expectation are unstable equilibrium points; stable equilibrium may appear in the course of acquisition and in the long term.

Unexpected Cases: As unexpected situations are far from negligible, related or similar unexpected events may further be acknowledged by the system since the expectation state is sensitive to such occurrences. The critical point is that known situations remain sufficiently important to handle expected situations by referring *globally* to past experience. However, unexpected instances are treated in relation with *particular* previous occurrences of the unexpected events.

Similarities Handling: Higher-level structures built by the aggregation module account for a certain *flexibility* of the system. When presented with cases close to previous cases, expected or unexpected, the system handles them with the methods employed in the preceding occurrences. This handling of related or similar cases might be considered as a primitive analogical processing [6].

7.6. Free-association and task-oriented structures

We have presented a modular system that encompasses a *task-oriented* problem-solving module *and* a *free-association* mechanism, brought together to account for a *learning* process. Such interconnected modules representing symbolic, numerical or both types of knowledge processing *subunits* may provide new computational models of brain functions. Task-oriented methods call for very definitite procedures, and many of the nervous system's structures are indeed made *in the image* of the task they perform. Higher functions though may be far more subject to change. Let us call *informative* a structure designed to perform a well defined task; higher functions correspond to the development of structures not yet as informative as innate task-oriented structures towards a more specific organization. The outside reality may well have the power to become progressively *imprinted* in each brain, establishing *knowledge reflexes.* Thus, structures accounting for free-association and other highly variable processes have a tendency to stabilize themselves in a certain configurations, and become task-oriented and informative. The generation of concepts is a malleable mechanism which nevertheless becomes more resistant to external variations with the system's experience. Human perception is likewise based on the confrontation of an internal representational state (expectation) and the reality; both sensitive and cognitive information might be thus processed. The development of new program architectures in Artificial Intelligence, from both the theoretical and practical point of view, should be a useful tool for modeling brain functions so as to reach a better comprehension of the neural code.

7.7. Prospects

7.7.1. Learning methods and future implementation

The current project was designed to evaluate the approach chosen in building a modular self-improving expert system. As results encourage further research based on this point of view, new modes of interaction between the dynamic memory and the problem solving module, based on first-look signs, are under consideration and currently under implementation.

The design of an adequate descriptive language for the clusters present in the dynamic memory, involving frames and scripts [39], will allow us to formalize the *analogy* implicitly described by a concept cluster [6]. Thus, this *strategy-oriented* structure super-imposed on the present dynamic memory would provide a deeper mode of interaction between the two modules.

The second mode of interaction is designed to get the maximum benefit from the first-look signs, and will run together with the preceding mode. First-look signs provide direct information on *how* to modify the contextual lists of relevant signs, altering the rule network according to experience. However the content of each individual rule is not to be modified in the process, following the principles of our approach. Use of the previously suggested descriptive language for updating the relevance lists induce a powerful imprint of past experience on the knowledge source structure as a network.

7.7.2. Control Structure Representation Language

The next step would be to provide a representation language to encode in a declarative manner rather than in a procedural way, the control structure for the first problem solving module [41, 12]. Such an implementation might yield to a closer interaction between the concept knowledge structure and the NCLOSE control structure.

7.7.3. Patient Evolution

The design of a rule base where RHS should embody the effects of a particular therapy on medical signs would account for a temporal simulation of the evolution of a patient. From the perception of such an evolution further information could be used in the KAA and KSI modules.

7.7.4. Explanatory Module

A program designed for effective use *must* be able to explain its own behavior and some powerful programs have already been written [45, 41]. At this point, two needs are to be considered in the present system. Generating explanations for the problem solving module, and generating explanations for the derivation of first-look signs from the dynamic memory.

Generation of explanations during or after the performance of the task may easily be done by using both the non-alterable knowledge sources and the current trace of execution kept by the system. However generating explanations for the behavior of the whole system, and specifically for the derivation and use of first-look signs, require the design of a new module. Using the previously described descriptive language for the dynamic memory as well as the ability to interact with the non-alterable knowledge sources, the concepts actually yielding first-look signs and the active influence of those in the search for initial hypotheses should be accounted for.

7.7.5. A Network of Knowledge Sources

As rule bases can be concatenated without loss of precision in such a system, larger areas of medicine could be considered by developing knowledge sources in different departments of an hospital, for instance. The control structure is able to investigate, if necessary, other aspects of a medical problem by switching to another knowledge base.

7.7.6. Critical signs and Plan Interaction

The problem of invasive investigations can be stated as following: the evaluation process should avoid asking questions referring to invasive investigations *before* having gone through more *routine* questions.

A multi-level system could be envisioned for handling such a difficulty. A higher level system would complete investigations before transferring control of the evaluation process to a different investigation level. Such a hierarchy should thus account for an underlying hierarchy in the order of investigations, or problem sub-spaces.

Another approach would be to perform a cyclic evaluation process, restraining in the first cycle queries to non-invasive investigations, then going through the different levels of difficulties in a sequential way. In this case one system is sufficient whereas in the other approach suggested here, as many systems are required as there are nodes in the hierarchy.

7.7.7. Modular medical intelligent systems

Intelligent systems in medicine are of central importance. The building of small, modular knowledge bases specialized in various domains should be a fruitful strategy. Such systems at first aim at *advising* the physician, handling or structuring areas of a problem. In later stages, small knowledge bases can be linked together. This modular approach allows easy contact between medical experts and computer scientists, and makes this research more readily available to both physicians and medical students. A project for building such systems is being currently initiated at the Hopital Necker-Enfants Malades in Paris.

7.3. Summary and Conclusion

The principal topic of this project was the study of the acquisition of expertise or *skill-refinement* by *learning from experience.* This question addresses the problem of increasing the performance of a given task by improving the use made by a problem solving module of a.given knowledge source. Production system formalism was chosen for the problem solving module, and clustering analysis and free-association methods were chosen for the aggregation and interface module. Hence this design leads to a system which is an alternative to systems acquiring knowledge by increasing or altering their rule base [13], An application in medical problem solving, consultation and advising, provided the ground for an evaluation of this approach.

The structure of the knowledge base was described along with examples of rules. A list of so-called relevant signs is associated to each rule, indicating a context in which it might be triggered, and which might be larger than the list of signs involved in the left-hand-side.

A detailed description of the rule-matching algorithm based on a formalization of the task of performing differential diagnostic operation and serving as the problem-solver module was presented. The various types of outputs were described, and an example of a session is presented in the appendices.

Two other modules that use these traces of execution to build a dynamic memory containing clusters of signs encoding the experience of the system were then described. These structures are used as a source of knowledge for improving the task performance. Clustering analysis and constrained search were the major tools for designing these modules.

This information actually represents the system's expectation. In the final section, results concerning the use of this expectation are described. The system acquires a new behavior by recognizing as accurately as possible hypotheses to generate that will be the input to the production system. Such a performance is comparable to pre-structuring the problem before solving it, an important feature distinguishing the expert from the non-expert in Medicine. Moreover, this behavior is enhanced when new situations are met. In effect, in order to behave more efficiently with regard to previously encountered situations, the system must actually learn about *new* situations, or otherwise it stabilizes.

When exposed to new cases the system reacts by modifying its expectation. This process somehow simulates a memory which allows recognition of known events even if those events were previously met only once, or twice. Furthermore, if the experience remains for long very different from such a past event, the latter might become unexpected again. Experiments involving two relatively important rule bases have shown that this approach yields correct medical results, as well as a satisfactory behavior with regard to skill refinement. We believe that this methodology encourages the design of a modular, learning expert system, based on larger knowledge sources and making use of the tools, ideas and prospects presented in this paper.

8. Acknowledgments:

We wish to thank Dr. Nicole Zygelmann-Athea for her dedication in working with us on this project. Productive discussions with Peter Szolovits, Allen Newell, Harry Pople, John McDermott and Mike Rychener have certainly influenced this work. Our thanks to David Servan-Schreiber and Ben Mulsant with whom we entered the AIM land, to Prof. Philippe Meyer, Dr. Robert Nahmias and Prof. Jean-Francois Boisvieux. Our grateful acknowledgments and thanks to the Centre Mondial Informatique et Ressource Humaine for providing us with computer time and disc space for the continuation of this research in Paris. We also wish to thank Dr. Scott Fahlman and the Spice Project group at CMU for the use of their computing facilities. And last but not least, we would like to express our most grateful thanks to Prof. Raj Reddy for providing us with the superb environment of the Robotics Institute and Computer Science Department of Carnegie-Mellon University, and to Dr. Mike Rychener for his intellectual critiques and support during the development of this research.

This research was partially supported by a grant (Weisveiller Grant Award, ATR) from the Association Claude Bernard - Assistance Publique, Paris. INSERM is the Institut National de la Sante et de la Recherche Medicale.

I. Problem-Solving example

```
Volunteered information about the patient:
        AGE 42
        MULTI-PARA yes
        IUD-PRESENTLY yes
        MENORRHAGIA yes

Now considering initial hypotheses :
(CYPROTERONE-ACETATE MICRO-PROGESTOGENS
CHECK-NO-HIGH-BLOOD-PRESSURE-NO-BENBMASTOP
EP-PROGESTOGEN-DOMINANT-CLIMATE CHECK-POSSIBLE-PROLACTINOMA
OTHER-METHODS MACRO-PROGESTOGENS INTRA-UTERINE-DEVICE MINIPILL
ESTROPROGESTOGENS-NORMAL-DOSES)
```

The number of initial hypotheses is very large, as the system has not yet acquired neither switch signs, nor any expectation or first-look.

```
Evaluating INTRA-UTERINE-DEVICE
Evaluating ESTROPROGESTOGENS-NORMAL-DOSES
Evaluating EP-PROGESTOGEN-DOMINANT-CLIMATE
Evaluating EP-PROGESTOGEN-DOMINANT-CLIMATE
Evaluating ESTROPROGESTOGENS-NORMAL-DOSES
Evaluating MACRO-PROGESTOGENS
*** Data needed : HISTORY-OF-MOTHER-SISTER-GENITAL-CANCER :  NO
Evaluating EP-PROGESTOGEN-DOMINANT-CLIMATE
Evaluating MACRO-PROGESTOGENS
Considering goal or subgoal -NORSTEROIDS-
Evaluating NORSTEROIDS
*** Data needed : WILL-TO-TAKE-ORAL-CONTRACEPTIVES :  YES
*** Data needed : HISTORY-OF-PHLEBITIS-VASC-ACC :  NO
*** Data needed : DIABETES :  NO
*** Data needed : HIGH-BLOOD-PRESSURE :  NO
*** Data needed : CHOLESTEROLEMIA :  1.8
Hypothesis or subgoal NORSTEROIDS is confirmed
*** Data needed : BENIGN-BREAST-DISEASE :  NO
Evaluating INTRA-UTERINE-DEVICE
Evaluating MICRO-PROGESTOGENS
Evaluating ESTROPROGESTOGENS-NORMAL-DOSES
Evaluating MICRO-PROGESTOGENS
Evaluating INTRA-UTERINE-DEVICE
Evaluating CHECK-NO-HIGH-BLOOD-PRESSURE-NO-BENBMASTOP
*** Data needed : IMPORTANT-STRESS :  NO
Evaluating MACRO-PROGESTOGENS
Evaluating INTRA-UTERINE-DEVICE
Evaluating OTHER-METHODS
Evaluating OTHER-METHODS
*** Data needed : RELIABLE :  YES
Evaluating CYPROTERONE-ACETATE
```

```
*** Data needed : IMPORTANT-ACNE :  NO
Evaluating CYPROTERONE-ACETATE
*** Data needed : HYPER-ANDROGENIA :  NO
Evaluating CYPROTERONE-ACETATE
Evaluating CYPROTERONE-ACETATE
Evaluating CYPROTERONE-ACETATE
Evaluating CYPROTERONE-ACETATE
Evaluating CHECK-POSSIBLE-PROLACTINOMA
Considering goal or subgoal -HYPER-PROLACTINEMIA-
Evaluating HYPER-PROLACTINEMIA
*** Data needed : IRREGULAR-MENSES :  NO
Evaluating HYPER-PROLACTINEMIA
Hypothesis or subgoal HYPER-PROLACTINEMIA is rejected
Evaluating INTRA-UTERINE-DEVICE
Evaluating MACRO-PROGESTOGENS
Evaluating MINIPILL
Evaluating ESTROPROGESTOGENS-NORMAL-DOSES
Evaluating ESTROPROGESTOGENS-NORMAL-DOSES
Evaluating OTHER-METHODS

 End of cycle.Current state of evaluation :
Considering goal or subgoal -CYPROTERONE-ACETATE-
Hypothesis or subgoal CYPROTERONE-ACETATE is rejected
Considering goal or subgoal -MICRO-PROGESTOGENS-
Hypothesis or subgoal MICRO-PROGESTOGENS is rejected
Considering goal or subgoal -CHECK-NO-HIGH-BLOOD-PRESSURE-NO-BENBMASTOP-
Hypothesis or subgoal CHECK-NO-HIGH-BLOOD-PRESSURE-NO-BENBMASTOP is rejected
Considering goal or subgoal -EP-PROGESTOGEN-DOMINANT-CLIMATE-
Hypothesis or subgoal EP-PROGESTOGEN-DOMINANT-CLIMATE is rejected
Considering goal or subgoal -CHECK-POSSIBLE-PROLACTINOMA-
Hypothesis or subgoal CHECK-POSSIBLE-PROLACTINOMA is rejected
Considering goal or subgoal -OTHER-METHODS-
Hypothesis or subgoal OTHER-METHODS is confirmed
Considering goal or subgoal -MACRO-PROGESTOGENS-
Hypothesis or subgoal MACRO-PROGESTOGENS is confirmed
Considering goal or subgoal -INTRA-UTERINE-DEVICE-
Hypothesis or subgoal INTRA-UTERINE-DEVICE is rejected
Considering goal or subgoal -MINIPILL-

Hypothesis or subgoal MINIPILL is rejected
Considering goal or subgoal -ESTROPROGESTOGENS-NORMAL-DOSES-
Hypothesis or subgoal ESTROPROGESTOGENS-NORMAL-DOSES is rejected
-----------------------------------------------

 End of cycle.Current state of evaluation :
Considering goal or subgoal -CYPROTERONE-ACETATE-
Goal or subgoal already evaluated and rejected
Considering goal or subgoal -MICRO-PROGESTOGENS-
Goal or subgoal already evaluated and rejected
Considering goal or subgoal -MACRO-PROGESTOGENS-
```

```
Goal or subgoal already evaluated and confirmed
Considering goal or subgoal -CHECK-POSSIBLE-PROLACTINOMA-
Goal or subgoal already evaluated and rejected
Considering goal or subgoal -MINIPILL-
Goal or subgoal already evaluated and rejected
Considering goal or subgoal -CHECK-NO-HIGH-BLOOD-PRESSURE-NO-BENBMASTOP-
Goal or subgoal already evaluated and rejected
Considering goal or subgoal -INTRA-UTERINE-DEVICE-
Goal or subgoal already evaluated and rejected
Considering goal or subgoal -OTHER-METHODS-
Goal or subgoal already evaluated and confirmed
Considering goal or subgoal -EP-PROGESTOGEN-DOMINANT-CLIMATE-
Goal or subgoal already evaluated and rejected
        <Considering goal or subgoal -ESTROPROGESTOGENS-NORMAL-DOSES-
Goal or subgoal already evaluated and rejected
-------------------------------------------------

 End of cycle.Current state of evaluation :
-------------------------------------------------
Evaluation performed with : 2 cycles, 12 questions asked.
23 nodes were visited.
```

STATE OF VISITED SUBGOALS :

```
Hypothesis PERIMENOPAUSE is rejected
Hypothesis NORSTEROIDS is confirmed
NORSTEROIDS was confirmed according to the rules :
(AND (NO DIABETES) (NO HIGH-BLOOD-PRESSURE) (> 3 CHOLESTEROLEMIA)
     (YES WILL-TO-TAKE-ORAL-CONTRACEPTIVES) (NO HISTORY-OF-PHLEBITIS-VASC-ACC)
     (YES RELIABLE))
Hypothesis ESTROGENS is rejected
Hypothesis ESTROGENS-ALLOWED is rejected
Hypothesis POSE-INTRA-UTERINE-DEVICE is rejected

Hypothesis NULLIPARE is rejected
Hypothesis TAKING-PILL is rejected
```

REJECTED HYPOTHESES

```
Hypothesis ESTROPROGESTOGENS-NORMAL-DOSES is rejected
Hypothesis MINIPILL is rejected
Hypothesis INTRA-UTERINE-DEVICE is rejected
Hypothesis CHECK-POSSIBLE-PROLACTINOMA is rejected
Hypothesis EP-PROGESTOGEN-DOMINANT-CLIMATE is rejected
Hypothesis CHECK-NO-HIGH-BLOOD-PRESSURE-NO-BENBMASTOP is rejected
Hypothesis MICRO-PROGESTOGENS is rejected
Hypothesis CYPROTERONE-ACETATE is rejected
Hypothesis HYPER-PROLACTINEMIA is rejected
Hypothesis MICRO-PROGESTOGENS-INDICATED is rejected
```

CONFIRMED HYPOTHESES

```
Hypothesis REMOVE-INTRA-UTERINE-DEVICE is confirmed
REMOVE-INTRA-UTERINE-DEVICE was confirmed according to the rules :
(AND (YES IUD-PRESENTLY) (YES MEMORRHAGIA))
Hypothesis MACRO-PROGESTOGENS is confirmed
MACRO-PROGESTOGENS was confirmed according to the rules :
(AND (YES NORSTEROIDS) (> AGE 40))
Hypothesis OTHER-METHODS is confirmed
OTHER-METHODS was confirmed according to the rules :
(YES RELIABLE)
```

II. Formalization

11.1. The symbolic proximity

For processsing the reasoning pathways, we used a proximity criterion analogous to a numerical distance in a metric space. As pathways are basically lists or sets, the Set Theory provides a mathematical background to assess properties of the aggregation algorithm. Reasoning pathways are expanded into sets of signs as explained before, and the proximity of two sets is computed as the symmetric difference between them. If A and B are such sets then :

$$[A3] = (AUB) - AnB$$

where u, n and - denote respectively union, intersection and difference.

From now on capital letters will denote subsets of the set of medical signs.

This proximity verifies these three postulates:

- [A,A] = 0,the null set
- [A,B] = [B,A]
- [A,C] ⊆ [A,B] U [B,C], where ⊆ denotes inclusion. This is the *triangle* inequality.

Let us list two useful properties, the proofs of which are easy and might be found in [20] for instance. If to every element n of an arbitrary set N we assign a pair of sets A_n and B_n, then :

- $[\cup A_n, \cup B_n] \subseteq \cup [A_n, B_n]$
- $[nA_a, nB_D] cn[A_n, B_a]$

where union and intersection are indexed over N.

II.2. Transitive Closure of a matrix

In this subsection we will introduce some notations for describing the process of transitive closure of a matrix. Here are some definitions :

- A relation from a set X to a set Y is characterized by a membership function

$$\mu : X \times Y \dashrightarrow E$$

  where E denotes $[0,+\infty[$ for a numerical approach, or the set of subsets of the set of signs for a symbolic approach.

- A relation can be equivalently represented as a matrix whose $(i,j)^{th}$ entry is $\mu(x_i,y_j)$.

- If R and S are two relations from X to Y and from Y to Z, resectively, then the composition of R and S, denoted RoS (or simply RS), is a relation from X to Z defined by :

$$\mu_{RS}(x;z) = +_y[\mu_R(x;y) . \mu_S(y;z)] \text{ with y in Y.}$$

  where "+" and "." denote dual operations such as max. and min, if $E = [0,+\infty[$, and union and intersection if $E = P(F)$ (F set of signs).

- The n-fold composition RoRo ...R n times is denoted R^n.

The transitive closure of a relation R is R^k where k is the smallest integer such that $R^{k+1} = R^k$. The elements of the matrix, or the membership function of this relation, are used to determine the clusters. This operation might only be performed on relations from a set to itself. We will denote the transitive closure of R by R^*.

For the notion of proximity we introduced in the last subsection, it can be noted that if R denotes the matrix obtained by computing proximities on a set of subsets, then $R = R^*$.

Let X be a subset of the set of signs, $X \subseteq P(F)$. Let R be a relation from X to X with membership function μ_R. Let x and y denote elements of X, then x and y belong to the cluster C_t iff :

$\mu_R*(x;y) \subseteq t$

Equivalently $C_t = \{(x;y) \in X \times Y / \mu_R*(x;y) \subseteq t\}$. Here t might be a positive real or a set of signs according to the chosen approach. In the next subsection we will point out some properties of clusters, specifically that they constitute a partition of X.

II.3. Clusters, modules and congruences

The process of aggregation of sets relies on the notion of congruence. We shall call two sets A and B congruent modulo M

$$A \equiv B\,(\mathrm{M})$$

if their symmetric difference $[A,B]$ belongs to M.

Before dropping explicit mention to M, we shall precise that in order to be complete, our definition of congruence must refer to M as a *σ-module*, which is a set of subset of the set of signs verifying :

- Every subset of a set of M itself belong to M.

• The union of a finite number of sets of M itself belongs to M.

We consider the subset of the set of signs F obtained by listing the elements of the final matrix of the KAA algorithm. From now on, the cr-module we will refer to for our congruence relation is the minimum a-module M containing F.

The congruence relation is an *equivalence relation,* reflexive, symmetric and transitive, thus allowing the partition of the set of signs into classes of congruent sets. Two classes are either disjoint or identical. A cluster is such a class. Each level of abstraction in the system is such a partition.

III. A control structure for rule-matching

It is now known that during the execution of the *recognize-act-cycle* of a running production system, the pattern matching operation is the most time and memory consuming, especially if there is a large number of rules and objects [17]. Two major ideas are to be developed in order to cope with this problem :

• *Pre-compilation,* or pre-processing of the rules in a network in order to propagate objects from the working memory as soon as they are created, altered or deleted. We thus avoid iterating the matching operation over the working memory at each cycle,

• *Differential selection* of rules at the beginning of each cycle, from the set of *nearly* fired rules of preceding cycle. We thus avoid iteration over the whole set of rules.

Applications of such ideas lead to efficient design of performant production systems [17,47], Our point here is that NCLOSE, with its differential diagnosis control structure, can be used as an alternative approach for the second critical characteristic stated before. Starting from an initial set of nearly-fired rules, the NCLOSE control structure allows fast retrieving of #//possible instances of the conflict set. This is done by going down to the conditions and its contained symbols, and retrieving the related conditions and rules by looking for those symbols in other rules. As this network of pointers from rules to conditions, then from conditions to symbols can be built at the beginning of the execution, the retrieving of the conflict set from the initial set of rules to be considered is easy. Thus the performance relies on die correctness of the initial set of rules, which is a first guess. If we assume that few objects will actually enter or get out of the working memory at each cycle, the nearly-fired rules of preceding cycle appears to be an excellent initial set to perform differential retrieving [11],

References

[1] Aiello, N.
A comparative study of control strategies for expert sytems: AGE implementation of the three variations of PUFF.
In *Proceedings of the AAAI-83.* AAAI, 1983.

[2] Berliner, H.J.
Search vs. Knowledge : An Analysis from the Domain of Games.
Technical Report CMU-CS-82-104, Carnegie-Mellon University, 1981.

[3] Buchanan, B.G., Smith, D.H., White, W.C., Gritter, F.J., Feigenbaum, E.A., Lederberg, J. and Djerassi, C.
Automatic rule formation in mass spectrometry by means of the Meta-DENDRAL program.
American Chemical Society 98, 1976.

[4] Bundy and Silver.
A critical survey of rule learning programs.
In *Proceedings of ECAI 1982.* ECAI, 1982.

[5] Carbonell, J.G.
A computational model of analogical problem solving.
In *Proceedings of the 7th IJCAI,* pages 147-152. IJCAI, 1981.

[6] Carbonell, J.G.
Learning by analogy : Formulating and Generalizing Plans from past Experience.
Technical Report CMU-CS-82-126, Carnegie-Mellon University, 1982.

[7] Carbonell, J.G., Michalski, R.S., Mitchell, T.M.
Machine Learning Part I : A Historical and Methodological Analysis.
Technical Report CMU-CS-83-135, Carnegie-Mellon University, 1983.

[8] Carbonell, J.G. and Minton, S.
Metaphor and Common-sense reasoning.
Technical Report CMU-CS-83-110, Carnegie-Mellon University, 1983.

[9] Chauvet, JM. and Rappaport, A.
Reflexion sur le Temps : Theorie des Realites.
Technical Report, Prix Scientifique Philips, 1980.

[10] Chauvet, J.M., Rappaport, A.
ALICE.
1982.
unpublished manuscript, Carnegie-Mellon University, Robotics Institute.

[11] Chauvet, JM.
PROSYL 2.
Technical Report, Ecole Polytechnique, France, 1983.
This thesis was presented as a partial fulfillment of the grade of Engineer.

[12] Clancey, W.
The advantage of abstract control knowledge in expert system design.
In *Proceedings of the AAAI-83.* AAAI, 1983.

[13] Davis, R.
Expert Systems: Where are we? And where do we go from here?
Artificial Intelligence Magazine 3,1982.

[14] Dietterich, T.G. and Michalski R.S.
A Comparative review of selected methods for learning from examples.
In Michalski, R., Carbonell, J., Mitchell, T. (editor), *Machine Learning,*. Tioga, 1982.

[15] Duda, R.O. and Hart, RE.
Pattern Classification and Scene Analysis.
Artificial Intelligence Group SRI, 1970.

[16] Elstein, A. S., Shulman, L. S.,and Spafka, S. A.
Medical Problem Solving.
Harvard University Press, Cambridge, MA., 1978.

[17] Forgy, C
A fast algorithm for the many pattern/many object pattern match object.
Artificial Intelligence 18,1982.

[18] Forgy, C. and McDermott, J.
OPS, A Domain-Independent Production System Language.
In *Proceedings Fifth IJCAl* pages 933-939. 1977.

[19] Forgy, C. L.
OPS5 User's Manual.
Technical Report CMU-CS-81-135, Carnegie-Mellon University, July, 1981.

[20] Hausdorff,F.
Set Theory.
Chelsea New-York, 1957.

[21] Keller, R.
A Survey of Research in Strategy Acquisition.
Technical Report 115, Department of Computer Science, Rutgers University, May, 1982.

[22] Kulikowski, C.A. and Weiss, S.M.
Representation of expert knowledge for consultation: the CASNET and EXPERT projects.
In Szolovits, P. (editor), *Artificial Intelligence and Medicine,*. Westview Press, 1982.

[23] Kulikowski, C.A.
Knowledge acquisition and learning in EXPERT.
In Michalski, R.S. (editor), *International Machine Learning Workshop,*. University of Illinois at Urbana Champaign, 1983.

[24] Lebowitz, M.
Generalization and memory in an integrating understanding system.
PhD thesis, Yale University, Department of Computer Science, 1980.

[25] Lenat, D. and Davis R.
Knowledge Based Systems in Artificial Intelligence.
McGraw Hill, 1982.

[26] Lenat,D.B.
The role of Heuristics in learning by discovery.
In Michalski, R., Carbonell, J., Mitchell, T. (editor), *Machine Learning,*. Tioga, 1982.

[27] Miller, R. A., Pople, H. R, Jr. and Myers, J. D.
INTERNIST-I, an experimental computer-based diagnostic consultant for general internal medicine
New England Journal of Medicine 307(8):468-476,1982.

[28] Mitchell, T. M.
Generalization as Search.
Artificial Intelligence 18,1982.

[29] Mulsant, B. and Servan-Schreiber, D.
A gentle introduction to artificial intelligence in medicine.
1982
unpublished manuscript, Carnegie-Mellon University, Robotics Institute.

[30] Nilsson,N.
Principles of Artificial Intelligence.
Springer-Verlag, 1982.

[31] Norman, D.A.
Analysis and Design of Intelligent Systems.
In Klix, F. (editor), *Human and Artificial Intelligence,*. North Holland Publishing Company, 1979.

[32] Patil, R.S., Szolovits, P and Schwartz, W.B.
Modelling Knowledge of the patient in Acid-Base and Electrolyte Disorders.
In Szolovits, P. (editor), *Artificial Intelligence and Medicine,*. Westview Press, 1982.

[33] Paulker, S.G., Gorry, G.A., Kassirer, J.P. and Schwartz, W.B.
Toward the simulation of clinical cognition: taking a present illness by computer.
American Journal of Medicine 60,1976.

[34] Pople, H., Meters, J., and Miller, R.
DIALOG a model of diagnostic logic for internal medicine.
In *Proceedings of the 4th IJCAI* pages 848-855. IJCAI, 1975.
The program was later renamed INTERNIST.

[35] Pople, H. E., Jr.
Heuristic methods for imposing structure on ill structured problems: the structuring of medical diagnostics.
In Szolovits, P. (editor), *Artificial Intelligence and Medicine,*. Westview Press, 1982.

[36] Langley,P.
PRISM user's manual
1980.

[37] Rappaport, A.
Closed Search: a hypothesis evaluator.
1982.
unpublished manuscript, Carnegie-Mellon University, Robotics Institute.

[38] Rychcner, M.D.
The Instructible Production System : A retrospective analysis.
In Michalski, R., Carbonell, J., Mitchell, T. (editor), *Machine Learning,*. Tioga, 1982.

[39] Schank, R.
Dynamic Memory.
Cambridge University Press, 1982.

[40] Schank, R.
The current state of A.I. : One Man's Opinion.
Artificial Intelligence Magazine 4,1983.

[41] Scott, C, Davis, R., Clancey, W. and Shortliffe, E. H.
Explanation Capabilities of Production-Based Consultation Systems.
Technical Report Stan-77-593, Stanford University, 1977.

[42] Shortliffe E. H.
Computer Based Medical Consultations: MYCIN.
American Elsevier, New York, 1976.

[43] Simon, H. A.
The Structure of ill structured problems.
Artificial Intelligence 4:181-201,1973.

[44] Sleeman D., Langley, P. and Mitchell, T. M.
Learning from solution paths: an approach to the credit assignment problem.
Artificial Intelligence Magazine 3,1982.

[45] Swartout, W. R.
Explaining and Justifying Expert Consulting Programs.
In *Proceedings Seventh IJCAI* pages 815-823. 1981.

[46] Szolovits,P.
Artificial Intelligence and Medicine.
In Szolovits, P. (editor), *Artificial Intelligence and Medicine,*. Westview Press, 1982.

[47] Waterman, D., and Hayes-Roth, F.
Pattern Directed Inference Systems.
New-York Academic Press, 1978.

[48] Winston, P. H.
Artificial Intelligence.
Reading, MA:Addison-Wesley, 1977.

[49] Winston, P.H.
Learning new principles from precedents and exercises.
Artificial Intelligence 18,1982.