Teaching Diagnostic Skills Using AI: an Architecture Suitable for Students and Teachers

Joël Courtois

Institut Supérieur d'Electronique de Paris
21, rue d'Assas 75270 Paris cedex 06 FRANCE

and
LAFORIA, Université PARIS VI
4, place Jussieu 75252 Paris cedex 05 FRANCE

Abstract

This paper shows how a new approach in the use of AI techniques has been successfully used for the design of an effective ITS in the domain of diagnosis training. The originality of this approach was to take into account three complex problems simultaneously: teaching diagnosis methods to students, giving the means to the teachers of maintaining the system by themselves and providing a tool easy to insert in the context of university laboratories. The architecture of the system is based on a distinct use of two kinds of knowledge representation. All the knowledge liable to modifications is gathered within libraries under descriptive forms easily maintained by the educational staff. General diagnosis knowledge independent of hardware, circuits and even application fields, is described with basic production rules and control metarules. The development of the system was based on the precise analysis of the expert's behaviour and of the user's needs, with the aim of making extensive use of the descriptive forms in order to minimize the static knowledge embedded in the rules. The system can work on a microcomputer and is used in an engineering school.

I. Introduction

Troubleshooting is a well-known domain of application for AI researchers. The challenge is to identify elementary physical components responsible for the malfunction of a more complex structure. The definition of an elementary physical component depends on the kind of maintenance that can be done. Thus, to estimate the troubleshouting complexity of a device it is always necessary to take into account the level of structural decomposition required for this purpose. Diagnosing a network of microcomputers is quite an easy task while troubleshouting analog circuits (De Kleer 89), (Dague 87) is a very complex problem not yet solved. Nowadays, having regard for a real life cycle (Courtois 90b), the insertion of a diagnosis expert system in an industrial context gives efficient results.

At the same time many research works were conducted on the domain of Intelligent Tutoring Systems. The results of these works are the following: first of all, it is necessary to be an expert on the studied field of application; a student model can be useful to adapt the behaviour of the system; teaching strategies must be studied and a userfriendly environment plays a significant part in the efficiency of the system. Teaching diagnostic skills includes the complexity of the field of application, as explained in Section II. However, even if these points require much more study, our aim is to prove that, nowadays, the state of the art allows the design of efficient ITS in a field of application where significant needs exist: Practical Works aid. Section III shows how, in order to overcome difficulties, the different points previously explained were taken into account for the SIAM system design. Another problem that is sometimes forgotten by ITS designers, is the teacher's point of view. It is a mistake to think that nowadays or tomorrow, ITS will replace teachers in any situation. The development of ITS implies the participation of the teachers and the first step, in the process of integration in the educational environment, is to conceive ITS as new technology tools. Consequently, Section IV explains how a true involvement from the teachers requires maintenance facilities, in order to allow them to improve the system by themselves, and how technical aspects must be taken into account for effective use in university laboratories. Section V describes the architecture of the SIAM system and the use of the different types of knowledge representations. The various models of description and expert modules used are shown with their intercommunications. An evaluation of the SIAM system is possible thanks to a real experimentation conducted over a period of one year. Section VI gives some findings of this experiment.

II. Complexity of Teaching Diagnosis Methods

At the same time as diagnosis systems were designed, research works on ITS were conducted within the same field of application. Three significant projects show us some steps of the evolution of the state of the art in the domain of teaching diagnosis:

- SOPHIE I, II and III (Brown 81);

In SOPHIE I, the student could take decisions and the inference procedure was good but the level of the explanations was weak. In SOPHIE II, the student was strongly directed and the inference procedure was limited but the level of the explanations was very good. The objectives of SOPHIE III were to improve these points by the design of three modules: an electronic expert, a troubleshouter and a coach.

- GUIDON I and II (Clancey 81-86);

GUIDON I was a general teaching system able to work with any knowledge base using EMYCIN. The pedagogical knowledge is clearly separated from the domain-specific knowledge. The difficulties came from the design of the technical knowledge bases. Only one diagnostic strategy was possible and the way it was conceived did not allow the student to memorize the expert rules. GUIDON II, with GUIDON-WATCH, GUIDON-DEBUG and GUIDON-MANAGE, improves the level of the explanations and focalizes on the teaching of the diagnosis strategies.

- SOCRATE (Moustafiadès 90);

SOCRATE is an expert system with a double vocation: SOCRATE-DIAGNOSTIC helps a technician on the spot when there is a breakdown, and SOCRATE-PEDAGOGUE teaches a rigorous diagnosis method by putting the student in an active learning context thanks to exercises drawn from real breakdowns. The domain concerns automatized production equipment. The aim is to teach a diagnosis method in an interactive and progressive environment thanks to various pedagogical and technical strategies.

These works show that to teach diagnosis strategies the following problems must be tackled:

- Design of an Efficient Diagnosis System.
- Explicit Identification of the Diagnosis Strategies.
- Explicit Identification of the Pedagogical Strategies.

III. Specificities of the P.W. Context

The domain chosen for experimentation is that of tutorials in higher education. This context is interesting from several points of view.

The technical level of the devices used is high but the complexity of the circuits done by the students is never really considerable. This allows one to use model-based and symptom-based approaches very efficiently. In this context, troubleshouting can be done with a good expert system.

The teaching of diagnosis strategies is one of the aims of the Practical Works. So these strategies are available from the "expert teachers". In the same way, the various levels of explanation that should be given to the students, in accordance with their level of knowledge, are known by the teachers. In this context the design of a student model is not possible (and not necessary) knowing that the interactions with the student are short and not regular.

Practical Works sessions make the student progress from the stage of beginner to that of expert while becoming aware of the links that exist between theory and practice. Practical assistance takes up most of the teachers' time since its aim is to solve the numerous problems that crop up during the manipulations. This practical assistance is relatively complex and we can distinguish three principal sources of problems:

- a misunderstanding of the work expected; this can come from incomplete knowledge of the underlying theory or of one of the materials involved;
- an error in the carrying out of the manipulation; this is the most frequent type of problem ranging from an assembly error to confusion about materials or to any possible mistaken use;
- a total breakdown or malfunctioning of a material; this is also quite frequent because of the bad handling to which the materials are frequently submitted.

Lastly, it is essential not to solve the problems for the student but rather to give them the knowledge and the know-how that will make them autonomous in a short time.

IV. Teachers' Point of View

The use of an Intelligent Tutoring System is a useful complement to the human tutors' work, for correct learning, in a practical context, needs very close surveillance to be efficient in a short time. But to be used in a laboratory the system must work on a large number of domains (physics, electronics, optics, etc.) and with several types of materials. Secondly, it needs the pedagogical skill which will allow the students to be autonomous as soon as possible. Lastly, it must be designed within a reasonable lapse of time because of the frequent modifications of the tutorial subjects linked to the evolution of the programs. With these conditions the following constraints appear:

- necessity for the system to be able to solve the problem with which it is confronted; this point requires the creation of a real expert in technical diagnosis;
- possibility of using the system for numerous tutorials in the same domain but also possibility of changing domain; the diagnostic knowledge supplied must be of a high enough level of generality to allow adaptation to several contexts;
- possibility of doing the maintenance of the system without AI technics or computer experts;
- a pedagogical approach in the search for solutions; the system must be able to explain itself and to react in an interactive way with the learner. It is not a question of solving the problem but rather of guiding the student towards the solution by the presentation of the methodology of research followed by the "expert teacher".

The most important consequence for the design of the system is that priority must be given to the definition of descriptives rather than to the writing of rules in order to dissociate the context-specific data from the procedures of data processing. The justifications of this approach will be found in the genericity of the technical diagnosis and the facilities of maintenance.

The technical expertise can be focused on the individual diagnosis of the materials. This choice can be discussed but we must not forget that we find ourselves in prepared sessions "that should work" and so we know the principal interactions between the materials. These interactions are therefore introduced as possible malfunctions at the individual level.

Finally the system must not be conceived to take the place of the human assistant but rather to work as a computerized complement.

V. SIAM Architecture

Introduction

The practical implementation of all the previously defined constraints is the SIAM system. SIAM is an expert system designed for diagnosis aid which guides the student providing him with explanations of the key steps of the expert diagnosis method. The system works on the spot, in a laboratory during Practical Works in Higher Education, and when real failures happen.

The SIAM development focalizes on the problems of reusability and maintainability of knowledge and on its use in a pedagogical aim.

Genericity of knowledge is an essential point in the reusability of expertise for the development of new applications. It also allows one to improve maintenance by separating general knowledge from knowledge specific to an application context. The pedagogical objective was taken into account right from the design phase of SIAM so that it has all the expert's behaviour justifications.

The SIAM knowledge and know-how are described, either with descriptive sheets designed using predefined models, or with production rules. All knowledge, related to domains or to hardware, liable to modifications, is gathered within libraries under descriptive forms easily maintained by the educational staff. Basic rules and control metarules which describe general diagnosis knowledge are independent of hardware, circuits and even application fields.

General Presentation

The general architecture of the SIAM system is proposed in the figure 1.

This architecture is based on the use of Models (a) and Descriptives (b) conceived from the models. The necessary descriptives collected together define the Context of Work (c) for the system.

SIAM technical knowledge can be divided into two principal categories: on the one hand, the descriptive knowledge, and on the other hand, the expert modules. Most of the system knowledge is represented in a descriptive way. The Production Rules (d), in the expert modules, are very general ones able to work with any descriptive made with the predefined models.

The Inference Engine (e) has no specific knowledge. It only activates the rules and metarules of the knowledge base.

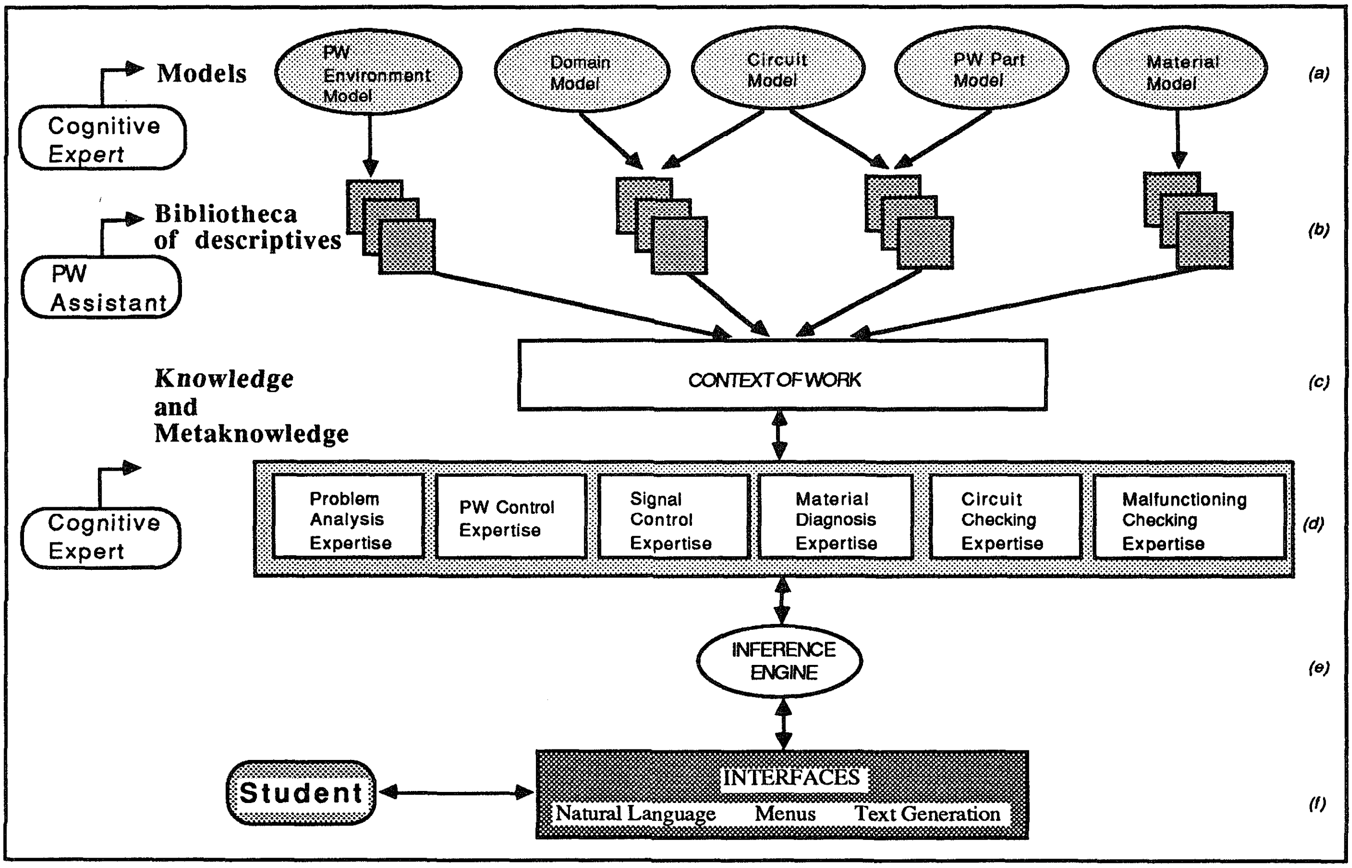


Figure 1: General architecture of the SIAM system

Interfaces (f) allow understanding and generation of natural language for communications between the Student and SIAM. The Menu technique is also used by the system.

Descriptive Knowledge

Before presenting the models used by the system it is necessary to indicate the kind of information they are able to represent. The models' attributes were designed according to the needs of representation. Descriptive knowledge given to the system through descriptives contains different types of information:

- conceptual descriptions;
- functional and relational descriptions;
- physical descriptions.

The first category supplies the most general knowledge, the highest abstraction level to which the expert's behaviour obeys. The extraction of these cognitive processes is often difficult because it comes up against the expert's judgement "it's obvious" or "it's logical". It is, nevertheless, at that level that we find the most useful knowledge, both to obtain a greater efficiency of the diagnosis system and a greater relevance of the pedagogical method being implemented. From the diagnosis point of view, identifying all the justifications of the expert's behaviour allowed us to conceive a general diagnosis strategy, independant of the technical field of application, the domain specificities being described within declarative descriptives. From the pedagogical point of view, these justifications were used to define steps of the diagnosis reasoning and to give explanations to the students. For example, we may find the conceptual description of a tutorial (what a student is going to do in a tutorial) and the conceptual description of a measure and a signal (we also had to describe the different types of signals which could be studied - light, electricity, motion).

The second category allows us to determine the function of the different elements which are used in the manipulations, as well as the main possible interactions due to the propagation of other elements' problems. Here the whole technical knowledge of the expert is thoroughly involved. In this category, we find the functional and relational descriptions of the devices. These descriptions of the various materials are made within the same frame (the Material Model) ; this descriptive constraint has the advantage of making it possible to write diagnosis rules quite dissociated from the materials and their application fields. Up to now, this structure has enabled us to describe all types of materials from the oscilloscope to the collimator as well as the lens and the simple wire.

The third category, the easiest to carry out, enables us to represent the various elements and their kinds of connections physically. Here we find the physical description of the materials (so as to detect the risks of confusion between some of them), the physical and functional description of the circuits (a list of the materials and the links between them in the circuit as well as the way the circuit is carried out and the way it can be controlled) and the description of the precise context of a tutorial problem.

The models which define the structure of representation of this knowledge are based on the following frames:

- Practical Work Environment Model: allows the representation of the objectives of practical work sessions. The principal attributes are PW_Success_Conditions, Measurement_Success_Conditions, Signal_Success_Conditions and Problem_Description. For examples: to succeed a PW session it is necessary to do correct measurements, good calculations and to understand the theory; to be able to do a correct measurement it is first necessary to have a correct signal.

- Domain Model: used for information related to the domain, the signal description in particular. As for the signal description, we have attributes like Name, Detection_Complexity, Measurement_Means, Origin and Transit.

- Material Model: used for all kinds of material from all domains. The most important attributes are Role, Directions_for_Use (with Setting, Non_Setting and Consequence), Test (with Symptom and Consequence), Limit (with Non_Respect and Consequence) and Physical_description. For each value of the attribut Directions_for_Use their is another attribut Setting with for each value another attribut Non_Setting and with for each value another attribut Consequence.

- Circuit Model: on the one hand, this model describes the physical circuit and on the other hand, its strategy of checking which is specific to the domain of application. No descriptives are made through this model, the corresponding information being included into the domain descriptives for the strategy of checking, and into the practical work part descriptives for the physical description.

- Practical Work Part Model: allows the description of each of the minor manipulations which constitute a session (e.g., materials, circuit, actions);

Bibliotheca of descriptives were conceived through those models in 3 domains of application, 50 materials and 40 parts of practical work. An example of representation of an electronic circuit is given in Figure 2.

A specific context of work for SIAM is made by gathering the required descriptives. With all these data, the system just needs the student's problem description to be able to begin its investigations thanks to the expert modules.

Expert Modules

The descriptive knowledge is used by a set of general expert modules thoroughly independent from the context, the fields of application and the materials.

These knowledge bases are managed by a large quantity of metaknowledge integrated into the various modules. On the one hand, the metaknowledge must control the local reasoning within the modules and, on the other hand, must provide the control transfer between the various modules.

The metaknowledge levels are not detailed and do not exist in a set way; the system itself chooses, at any given time, to which level the control is transmitted dynamically and without any pre-existing order.

The principal expert modules are the following ones:

- Tutorial Control Expertise; this high level expertise tries to verify if each of the necessary conditions for a correct carrying out of the practical work is present. It principally uses the conceptual descriptions. From this level, metaknowledge can transfer the control to the theory control expert module or to the calculus control expert module or to the measurement control expert module;

- Problem Analysis Expertise; from information supplied by the student, this expertise tries to deduce the largest possible quantity of data on the real state of the manipulation. From this level, the control is transmitted to the tutorial control expertise;

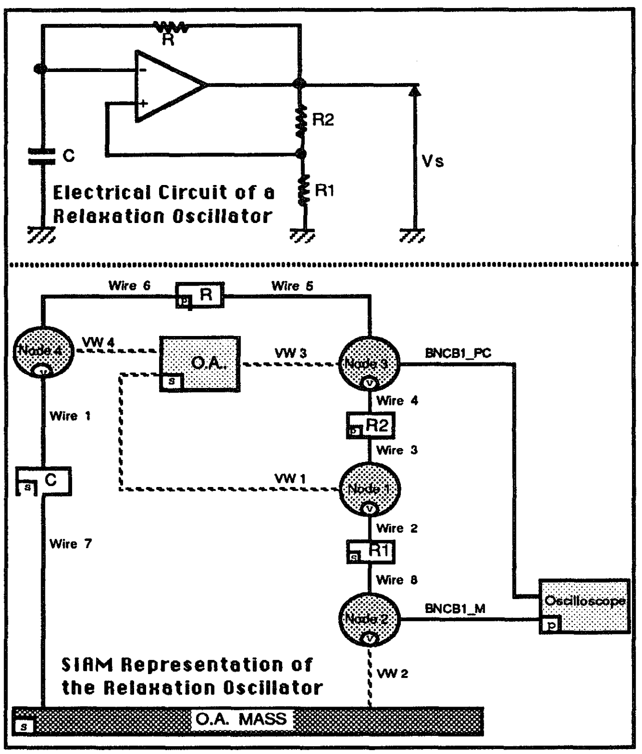
- Signal Control Expertise; this high level expertise will be activated when the functioning of the studied signal seems to be in doubt. From this level, metaknowledge can guide investigations according to three axes:

  . signal existence problem;

  . signal state problem;

  . signal value problem.

- Material Diagnosis Expertise; this works following several steps directed by metaknowledge: finding of the whole set of defective settings, organization of these causes according to their complexity, dialogue with the student making checks, troubleshooter tests, control of the borderline cases of use. The main rule applied in this expertise is:



KNOWING: the problem is X;
there exists a device A,
whose *Consequence*
of the *Non_Setting*
of a *Setting*
of the *Directions_for_Use*
is X;

THEN: the device A is open to be the source of the problem.

- Circuit Checking Expertise; this expertise will enable the student to examine and thus to control his circuit. The way of control to be followed is explicitly found, step by step, in the domain description thanks to the circuit description attributes.

- Malfunctioning Checking Expertise: the order of control of the devices is deduced from their rate of breakdowns and from their difficulty of testing.

Some of these modules contain several levels of metaknowledge. The various expert modules must operate as if they were autonomous agents able to transmit the control to each other, each time it is necessary.

Interfaces

Different interfaces are necessary to give more userfriendliness in the relations between the student and the system.

The first contact is made with a natural language interface, based on an ATN (Woods 70), that allows the student to explain the problem which occurs. In order to avoid a situation of misunderstanding, if some trouble appears with this interface, another one based on the technique of keywords and menus is used as a back-up.

MATERIAL	MANIP	=	OA
DESCRIP	OA	=	OPER_AMPLIF
CONNECT	OA	=	SERIES
MATERIAL	MANIP	=	R1
DESCRIP	R1	=	RESISTANCE
...	...	...	...
LINK	OA	=	VW1
LINK	OA	=	VW4
LINK	OA	=	VW3
LINK	NODE1	=	VW1
LINK	NODE1	=	WIRE2
LINK	NODE1	=	WIRE3
LINK	R1	=	WIRE2
LINK	R1	=	WIRE8
LINK	NODE2	=	WIRE8
LINK	NODE2	=	VW2
LINK	NODE2	=	BNCB1_M
LINK	OA_MASS	=	VW2
LINK	OA_MASS	=	WIRE7
...	...	...	...

Figure 2: Part of the circuit descriptive

During the session, the explanations given by the system and the questions asked the student are dynamically conceived thanks to a specific interface that uses predefined pieces of sentences linked to the objects manipulated by the system: rules and attributes. The general structure of the sentences is defined with some special actions contained in the rules. The specificities of a context are found in the messages linked to the descriptions of the domain, circuit, materials, etc. The possible answers to the questions asked by the system are given in a menu which is also dynamically conceived.

At the end of a session, the student can give some comments which contain technical indications or appreciations concerning the SIAM system behaviour.

VI. SIAM Evaluation

The system presented above has been evaluated over one year (1988-1989) in one of the laboratories of the Institut Supérieur d'Electronique de Paris (France), where students can use it in three domains, optics, electronics and electrotechnology. This real experimentation has been very valuable for the final touches to the system.

The way the system works is satisfactory and corresponds well enough to the expert's behaviour. The student is able to catch the whole cognitive process followed by the expert which enables him to understand his future problems better (examples of diagnosis sessions can be found in (Courtois 90a)).

It is important to note that the student is normally seldom supplied with this kind of information in tutorials, because the assistant usually explains why it did not work once the solution is found out, but does not systematically specify how he has detected the origin of the problem (whence the experts' magical aspect).

The amount of explanations which do not depend on any artificial model of the student, must be precisely adapted so that the students should not consider the system as too talkative or too fast. During the system development, it was necessary to condense some descriptions which appeared to be too detailed or, on the contrary, some others had to be more detailed, so that the precise origin of the problem could be found faster.

The natural language interface must contain no errors because users would get tired of it very quickly. A graphic interface could improve the userfriendliness even more for the circuit checking.

Finally the system fulfils its last aim which is to create new tutorial subjects rapidly. As a matter of fact, modification or creation of a practical work subject, pedagogical aspects included, can be done and tested in a very short time: if all the devices needed are already described, it will take about one hour of work to describe a new part of a practical work session.

VII. Conclusion

The whole system's knowledge, descriptions and expertise, is stocked within a single format close to SNARK's (Laurière 88), that is to say, a production-rule-based system in predicate logic. This single representation enables the system to handle, in the same way, independent facts, structured descriptions, rules and metarules. SIAM can work on microcomputers and then be easily introduced in university laboratories where microcomputers are available. In this thesis research work, the adopted point of view consisted in giving the maximum of explicit knowledge to the system, so that it could control the development of its actions itself due to a large quantity of metaknowledge. The model-based representation of the most important part of this knowledge and the conception of general production rules are the principal characteristics of this work. They both offer the opportunity of a reusability of knowledge and a facility of maintenance that allows the teachers to monitor the evolutions of the system themselves.

The obtained results have justified these choices both in the system's efficiency, possible evolutions and adaptations, and also pedagogically speaking. In a domain where real needs exist, this work shows that the development of ITS based on the SIAM architecture is an available answer.

References

Brown, J.S., Burton, R.R., De Kleer, J., 1981, *Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II and III*. Xerox Palo Alto Research Center.

Clancey, W.J., Letsinger, R., 1981, NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching, *7th IJCAI 81*, p. 829-836.

Clancey, W.J., 1986, From Guidon to Neomycin and Heracles in Twenty Short Lessons, *ONR Final Report 1979-1985*. Stanford Knowledge Systems Laboratory.

Courtois, J., 1990a, SIAM: a System that Adapts Easily to New Domains and which Teaches its Method, *Thèse de Doctorat d'Université*. Université Paris VI.

Courtois, J., Moustafiadès, J., 1990b, Human Being and Expert System: Which One Uses the Other ?, *COGNITIVA 90*. Madrid.

Dague, P., Devès, P., Raiman, O., 1987, Troubleshooting when modeling is the trouble, *AAAI 1987*. Seattle.

De Kleer, J., Williams, B.C., 1989, Diagnosis with Behavioral Modes, *11th IJCAI 89*, p. 1324-1330.

Laurière, J.L., 1988, *Intelligence Artificielle (tome2) Représentation des connaissances*. Eyrolles. Paris.

Moustafiadès, J., 1990, *Formation au diagnostic technique: l'apport de l'intelligence artificielle*. Masson. Paris.

Pitrat, J., 1990, *Métaconnaissance, Futur de l'Intelligence Artificielle*. Hermès. Paris.

Woods, W.A., 1970, Transition network grammars for natural language analysis, *Communications of the ACM* vol.13.