

Reutilization of diagnostic cases by adaptation of knowledge models

B. Chebel-Morello, M.K. Haouchine, N. Zerhouni

Automatic Control and Micro-Mechatronic Systems Department, FEMTO-ST Institute, 24, Rue Alain Savary, 25000 Besançon, France
{Brigitte.morello, karim.haouchine, noureddine.zerhouni}@ens2m.fr}

Abstract.

This paper deals with design of knowledge oriented diagnostic system. Two challenges are addressed. The first one concerns the elicitation of expert practice and the proposition of a methodology for developing four knowledge containers of case based reasoning system. The second one concerns the proposition of a general adaptation phase to reuse case solving diagnostic problems in a different context. In most cases, adaptation methods are application-specific and the challenge in this work is to make a general adaptation method for the field of industrial diagnostics applications. This paper is a contribution to fill this gap in the field of fault diagnostic and repair assistance of equipment. The proposed adaptation algorithm relies on hierarchy descriptors, an implied context model and dependencies between problems and solutions of the source cases. In addition, one can note that the first retrieved case is not necessarily the most adaptable case, and to take into account this report, an adaptation-guided retrieval step based on a similarity measure associated with an adaptation measure is realized on the diagnostic problem. These two measures allow selecting the most adaptable case among the retrieved cases. The two retrieval and adaptation phases are applied on real industrial system called Supervised industrial system of Transfer of pallets (SISTRE).

Keywords: case-based reasoning, adaptation, adaptation-guided retrieval, dependency relations, hierarchical model, context model, diagnostic.

1 Introduction

This study is a part of our research work realized in the European project SMAC. The goal of this project was to develop a distributed cooperative knowledge oriented platform that proposes a set of maintenance services (Karray et al, 2009).

These services offer support to the maintenance operators in their daily tasks. The platform is composed of several modules like the equipment model for instance, and a functional analysis and resource management unit that is interconnected by domain ontology (Karray et al, 2011) to ensure an intelligent maintenance policy within a company. The knowledge models are used in this platform to solve practical problems such as fault detection, fault diagnosis, etc. in industrial applications.

The paper aims at building an intelligent application based on a case based reasoning system that is dedicated to industrial diagnostic and repair in the context of maintenance services.

Althoff (Althoff et al, 1996) assert that case based-reasoning (CBR) is the most appropriate technology to implement a knowledge based system. Moreover, CBR is frequently proposed as a methodology for knowledge management applications, in particular in experience based system technologies. It presents the expert knowledge as past and real experiences that are easily understandable by human users. It is a problem solving and learning method that uses similar past problems formalization to solve new ones by adapting them to the context. This method is suitable for diagnostic applications, because fault diagnostic is a domain based on the experience of human experts, where problems are recurrent and can be reused. In addition, diagnostic by case-based reasoning is one of the methods preferred by the industrials. Indeed,

reasoning in this method, and unlike machine learning methods where the learning phase has a significant cost and requires time, can start with an incomplete case.

Therefore designing a diagnostic knowledge oriented system requires time and availability of experts to explicit their practice. However, the cost of this service is very high, and thus this phase needs to be optimized. Indeed, the challenge in knowledge management is the knowledge gathering. AI Jacobson and L. Prusak(1996) studied the knowledge management in organizations, their prediction on the knowledge cost for the year 2010 was € 2.6 billion. Their statistical study involved more than 200 knowledge workers of different organizations. **Fig. 1** shows that 37.7% of the time of knowledge workers is dedicated to the elicitation phase and 45.9% to the knowledge adaptation phase.

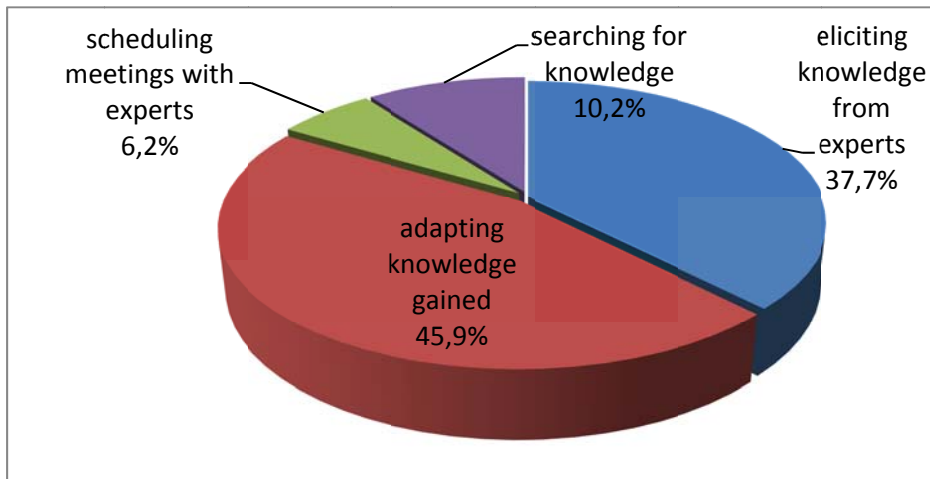


Fig. 1: Percentage of knowledge workers ‘time spent (Jacobson, A., Prusak, L. 1996)

Our problem is to develop a knowledge based diagnostic system for SISTRE a supervisor industrial system of pallet transfer (see **APPENDIX A**) while minimizing development costs, this can be achieved thanks to the use of an existing method. To reuse an adequate method, we briefly review some of the CBR industrial diagnosis system.

Industrial systems developed in the CBR diagnosis, can vary from a car studied in a diagnostic technique system named CREEK (Aamodt, 2004) to a locomotive defined for remote diagnostics (Varma, 1999), to aircrafts (such as Boeing 747 aircraft developed in CaseLine (Watson et Marir, 1994) and used as a demonstrator by British Airways or Boeing 737 in CASSIOPEE (Bergman et al, 2003) owned by CFM International). There are also industrial printers studied by Domino UK Ltd in CHEKMATE and presented in (Grant et al,1996), gas turbines studied in a fault system of General Electric Energy in Atlanta (Devaney et Cheetham, 2005) and complex machine like Patdex (Richter, 1991) and Nodal CBR (Cunningham,1994). There is a wide variety of methods ranging from classification problems when there is just a weak domain theory, to knowledge based systems. More knowledge-oriented systems such as Gas Turbine, Creek, Cassiopee, Pad'im, NodalCBR, Patdex, use models of knowledge different from each other. In most systems, a case characterizes a diagnostic experience.

We note that there is no common method for building a CBR system. The construction depends crucially on the representation of cases and knowledge models of the application domain. That is phases of CBR cycle. Actually, authors provide us just with the cases representation of the system and without any associated methodology.

Very few authors defined the design methodology in the case based diagnosis system. Lamontagne & Lapalme (Lamontagne, 2002) represent a generic model of case based

reasoning system in which they combine the CBR online cycle composed of 4 phases (elaborate, retrieve, reuse, retain) with knowledge container defined by Richter (Richter, 1995) (vocabulary; case base; similarity metrics; adaptation knowledge) making it possible to preserve and exploit the past experiences.

In the same way our team (Rasovska et al, 2008) combines the CBR reasoning “capitalize” and “actualize” phases with the knowledge “detection” and “preservation” in the knowledge capitalization cycle defined by Grundstein (Grundstein, 2000). The elicitation step is not defined by interviews with experts but by a methodology based on engineering safety tools, and reliability concepts such as FMECA and failure history (Rasovska et al, 2008). The creation of knowledge models is carried out by the analysis of the equipment and its decomposition (approach modeled on the practice ground of Cegelec company).

The idea developed in this paper is similar to Rasovka’s methodology, where a CBR system is built from knowledge models used by maintenance experts like functional and dysfunctional equipment models that are accessible on all equipments and easy to build when not available (Rasovska et al, 2008). This choice minimizes the cost of experts.

To elaborate an oriented knowledge CBR system, we must conceive first the different knowledge containers of CBR and secondly the CBR reasoning phase.

- (i) In the knowledge elicitation, our proposal is to build
 - a. An appropriate case base (a case formalization based on the definition of the diagnostic taking into account the indicators used by the maintenance users).
 - b. Similarity measures taking into account missing values and adaptation measures reflecting the adaptation cost
- (ii) In the reasoning process, our proposal is to develop a reasoning process suitable for this model. The reasoning process is dedicated to retrieve the similar case and adapt it to solve problems in other contexts. This adaptation phase is complex and is usually designed for a specific application.
 - a. The first challenge in this reasoning phase is to propose an algorithm based on the model and independent of the specific application.
 - b. The second challenge is to minimize the adaptation cost in the retrieve and reuse phases by proposing an “adaptation-guided retrieval phase”.

Our objective in this work is to minimize the cost of designing the knowledge diagnostic system. To evaluate the design cost of our diagnostic method we define in section 6.4.2 three indicators; the adaptability of the method, the reuse cost and the creation effort.

The paper is organized as follows. We first describe the principle of the elicitation approach which defines the knowledge containers of CBR system. In section 3, we focus on the retrieve step of the reasoning phase. One might think that the most similar case is always the best candidate. However, the literature reviews of the retrieve phase show that this is not always the case (Smyth et Keane,1993, 1995, 1998; Cordier et al, 2006). Consequently, we propose an adaptation-guided retrieval method applied to the industrial diagnostic based on two measures: the first one is similarity and the second one is adaptation. Section 4 deals with the adaptation phase, which is applied to the best adaptable retrieved case.

To define a general adaptation method on symbolic data in the field of industrial diagnostic, we are interested in the adaptation approaches in all applications of CBR.

Some studies of the “memory-based reasoning” (Kasif, 1995) avoid this step because the wealth of the case-base can compensate for the adaptation phase (Standfill, 1996).

However, other authors, like us, develop this phase to enrich the case-base. In this context the adaptation step is the core of CBR (Chebel-morello et al., 2011; Lieber, 2007). Furthermore, prior works on adaptation were dedicated to a given application. To avoid this specificity, three axes have been explored: (i) Adaptation Knowledge Acquisition (AKA) aims to define general principles of clarification in the studied field. A complete state of the art concerning these methods can be found in (Lieber, 2007). (ii) Catalogs of adaptation strategies are applied in several domains and are given in (Riesbek,1989). (iii) Unifying approaches are studied in order to find a general adaptation model as proposed by Fuchs (Fuchs et al, 2000). These authors propose a general adaptation algorithm independent of the application scope. It is applied to digital data as an interval that can be extrapolated. Therefore, we were inspired by the unifying approaches of Fuchs (Fuchs, 2000).

We propose a method is based on the dependencies between the problem and the solution of a solved case and uses two knowledge models. Three relations of dependencies are defined and exploited to adapt a retrieved case within an adaptation algorithm described in the same section. The matching carried out at the time of the retrieval, combined with dependency relations between the problems and solutions, can adapt the solution to the target problem.

Section 6 is devoted to the evaluation of the methodology of our CBR diagnosis system with the other industrial diagnosis systems.

2 Elicitation knowledge approach

2.1 Introduction

Our ambition is to construct a diagnostic system based on the European standard definition that specifies diagnostic as “they are actions carried out to detect breakdowns, localize them and identify the cause. (Maintenance terminology, 2001):NF EN 13306 of diagnostic. In order to build this system of help of diagnostic, we decompose the equipment into zones that include components. A zone will be characterized by the descriptors d_1 d_2 d_3 (these descriptors are necessary to define the existing zones).

Within each zone, each component has a specific function. If this function is not fulfilled, it will induce dysfunctions that are revealed by some symptoms and characterized by the state of the component and its functioning mode. **Fig. 9** gives you the internal functional analysis of an assembly station in SISTRE.

To define the functional mode d_i^{FM} , the observed state is compared with the expected state in functional context and identified to the normal (nor) or abnormal (abn) mode.

Those components are characterized by symptoms descriptors. Three parameters are defined by: name, measure defining its state and functional mode $d_i = (d_i^{value}, d_i^{state}, d_i^{FM})$

We describe the failure that might occur using the problem descriptors that include zone’s descriptors, descriptors in relation to the functions and/or dysfunctions associated to the zone where the failure is taking place.

These components will be classified into different families of components in a hierarchy of components (family of sensors, pneumatic actuators). This will be used by the reasoning phases, mainly, the phase of identification of the failing component.

The localization of the failed zone, when a fault is detected, is done thank at a conceptual graph (see Fig.3a and Fig.3b) where the node is the value of the zone descriptor and the solution is the failed zone. Failed zones are composed of the components potentially failing.

A case will describe the experience of how a diagnostic problem is solved. A retrieval phase will find the cases the most similar to the case to be solved by using similarity measures. An adaptation measure will select the most easily adaptable case among the most similar cases. An adaptation method will be applied if no identical cases are found in the case base. The method is based on the relations of dependence between the variations of the problem and the variation of the solution. It includes 3 classes of adaptation that will identify the cause of the failure in function of the context of the breakdown.

2.2 Diagnostic case representation

The case base reflects the experiment of solving a fault by linking the dysfunctional mode of component to the cause of this fault, and the repair action. Indeed, we propose the following representation.

A case is composed of a Problem and a Solution part:

Case= (d₁, d₂, .d_i, D₁, D₂, D₃, D₄)

- (i) The problem space of the case contains descriptors relative to both the **localization** and the functional part of the equipment.
- (ii) In the solution space of case, descriptors retain information about the detection of failure and the identification of the cause.

A case is formalized by two kinds of problem descriptors: the localization descriptor and the symptom descriptor, and by 4 solution descriptors: D₁: the Class detected, D₂: the cause of failure, D₃: the Repair action, D₄: the zone of failure.

The existing class is defined in the components taxonomy model.

To simplify the notation of the source case descriptors we note $d_i (S_l) = ds_i$
and the target case descriptors $d_i (T) = dt_i$

$$d_i (S_l) = (d_i^{value} (S_l), d_i^{state} (S_l), d_i^{FM} (S_l)) = (ds_i^{value}, ds_i^{state}, ds_i^{FM})$$

$$d_i (T) = (d_i^{value} (T), d_i^{state} (T), d_i^{FM} (T)) = (dt_i^{value}, dt_i^{state}, dt_i^{FM})$$

The solution part is composed of four descriptors, the first one is relative to the class of the fault component, the second is dedicated to the component causing the failure, the next one describes the repair actions, and the last one defines the zone of the failure.

The notion defined in this paper will be d by the application SISTRE:

For example: let us consider the puller, a specific component in equipment. This component can have two states linked to its position: [front; back] and can have two functional modes [nor, abn].

The descriptor associated to the puller in a diagnosis case can be written as illustrated in Table 1:

Table 1: example of source case

Problem part						Solution part				
Localization		Supervisor				Class	Cause	Repair	Zone	
ds ₁	ds ₂	-----	ds _i ^{value}	ds _i ^{state}	ds _i ^{FM}	-----	Ds ₁	Ds ₂	Ds ₃	Ds ₄
internal Ring	entry	-----	puller	front	abnor mal	-----	Electrical actuator	Blocked S1	unblo ck	Entry internal Ring

We can see in Table 1 a case containing in the supervisor descriptor the first symptom relative to the pusher ds₃.

$$ds_3 = (ds_3^{value} = puller, ds_3^{state} = front, ds_3^{FM} = abnormal)$$

The localization of this equipment is the entry of internal Ring (ds₁ and ds₂) and so on. The solution is composed of a class of failure Ds₁=The Electrical Actuator and the cause of failure Ds₂ = Stopper S₁ blocked. Ds₃ is the repair action and Ds₄ the Failure zone.

2.3 Knowledge models associated with the Case

Moreover, the knowledge representation is based on two models associated with the case-base, namely: the context model and the components taxonomy model.

2.3.1 The components taxonomy model

A case has a formalization object and defines a hierarchy of descriptors containing both problem and solution (Haouchine et al, 2008).

The model is determined from the functional components analysis of the industrial plant. Every group of components (in the source case or observed case) is regrouped by functional classes, and constitutes a component's hierarchy which is common to the problem descriptors "ds" and solution "Ds". A part of SISTRE hierarchical model of components is described in Fig. 2

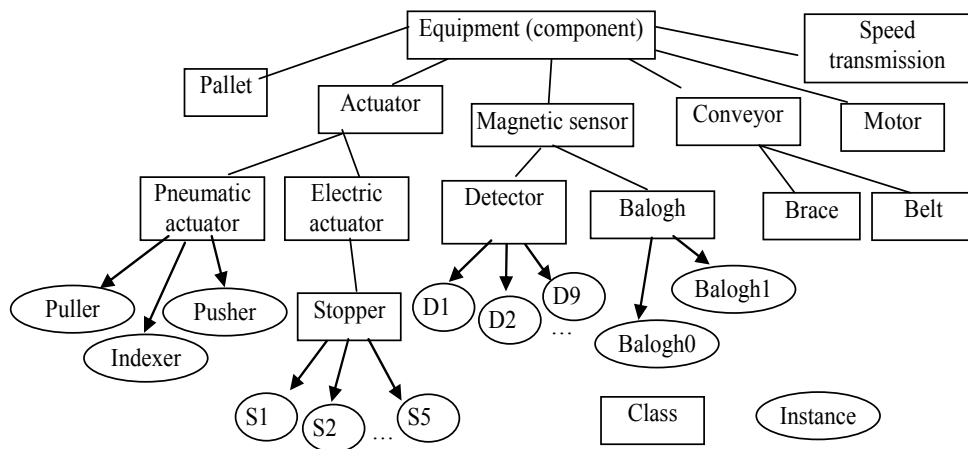


Fig. 2: A part of the SISTRE's components hierarchy.

2.3.2 Context Model

The context model is a contextual graph allowing the localization of components comprising a failure and selection of concerned components compared to the set. Therefore, the context model enables to inform the “localization” descriptors in order to determine the failure zone and the components potentially failing. The course of a pallet will be followed. Using a contextual graph, as shown in Fig. 3a, components likely to be failing will be localized.

An example of a context model concerning the descriptor “Ds₁” is shown in Fig.3b.

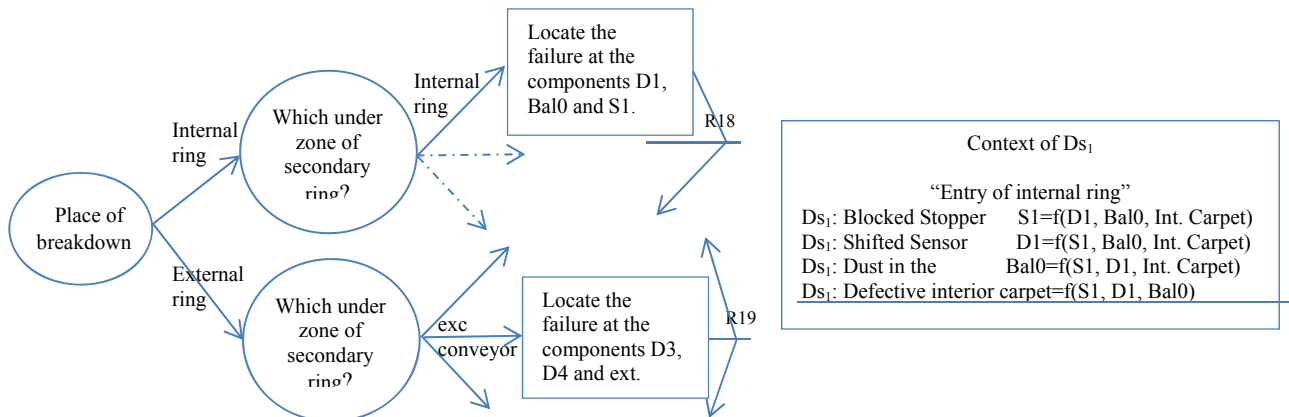


Fig. 3a: A part of contextual graph of the equipment. Fig. 3b : A context model of “Ds₁”

The context allows the localization of components problems and the selection of the right descriptors compared to all others. Therefore, these present components constitute the context in which the failing component is identified. A dependency relation is associated with these components. The descriptors of the localization part are exploited by a context model in two phases of the CBR. In the phase of elaboration of the target case, the user is asked a dynamic tree of questions, and retrieval phase selects the correct element to be substituted in adaptation phase.

2.4 The diagnostic case base

A set of 125 cases is considered in the SISTRE case base which corresponds to a part of the component malfunction. The cost to realize the exhaustive case base becomes too large in complex equipment, and to get to this industrial condition, we do not exhaustively list malfunctions of all components. The case problem part is composed of seven descriptors. The first two descriptors define the localization of the failure. This localization is determined by “ds₁: zone”, “ds₂: pallet site”.

Let us consider the example of case S1 (Source 1 or observed case 1) in the Table 2.

Table 2: A part of the SISTRE case base

Index	Problem															Solution					
	Localization		Fonctionnel part													D1	D2	D3	D4		
	d1	d2	d3			d4			d5			d6			d7						
Zone	Emp palette	Dét prin	Etat	MF	Act Pneu	Etat	MF	Act Elec	Etat	MF	Dét de pré	Etat	MF	Dét mag	Etat	MF	Failure Class	Identification cause	Repair action	Failure zone	
S1	Internal Ring	entry	D1	0	abn				STP1	top	nor				Bal0	1	nor	Presence sensor	Shifted D1	Replace	entry Internal Ring
S2	Internal Ring	entry	D1	1	nor				STP1	bot	abn				Bal0	1	nor	Electric actuator	Blocked S1	Unblock	entry Internal Ring
S3	Post Zone		Bal1	1	nor	Ind	top	an	STP4	top	nor							Pneumatic actuator	Obstacle under indexer	Remove object	Post Zone
S4	Internal Ring	exit	D8	1	abn	Puller	down	nor	STP6	top	nor				Bal1	1	nor	Presence sensor	Shifted D8	Replace	exit Internal Ring
S5	External Ring	Ext conv	D6	1	nor				STP5	top	nor	D5	0	nor	Bal1	1	nor	Pneumatic actuator	Blocked puller	Unblock	Ext conv exit ext Ring
S6	Post Zone	Ind							STP4	bot	abn	D6	0	nor				Magnetic sensor	Strong magnetic field	Clean Balogh	exit External Ring
S7	External Ring	Ext conv	Bal1	0	nor	Pusher	down	nor	STP5	top	abn	D7	0	nor	Bal1	1	nor	Electric actuator	Blocked puller	unblock	exit External Ring

This case represents a problem in the detector “D₁”.

The localization part indicates that there is a failure on the entry of the “principal” internal ring. Then, the supervisor part provides the components state involved at the failure zone. The STP1 stopper is in “top” position which has a “normal” functional mode. The balogh ”bal0” has value “1”, which means that it must enter the working area so that it can be treated by a robot. Finally, the D1 detector does not detect the presence of the pallet which is in abnormal mode (“abn”).

The solution part is made up of, the class descriptor of the failing component, a descriptor identifying the failing component, the repair action and the failure zone.

3 Reasoning process: retrieval phase

In this work, three phases, elaboration retrieve and reuse compose the reasoning process (Mille, 1999). Case descriptors of the localization part are exploited by a context model (see Fig. 3a) in

- (i) The elaboration phase. The user answers a dynamic tree of questions in order to build a target case (case to solve).
- (ii) The adaptation phase. The correct element is selected to be substituted in the adaptation algorithm.

In this section, the retrieval phase is broached.

3.1 Retrieval phase

The main difficulty in the retrieval phase is the choice of the right selection criterion for cases to be retrieved (Althoff et Bartsch-Spörl, 1996). Most of the traditional approaches use simple similarity measures, while others involve the organization of the case-base to

improve the retrieval quality (Cordier, 2008). Thus, the retrieval phase can be described by two categories: the first is “simple retrieval” and the second is “combination retrieval/adaptation”. The current study is focused on the second type.

Before the Nineties, the two phases of retrieval and adaptation were used in a completely independent way until Veloso (Veloso, 1992) brought new life and suggested the combination of these two steps. Smyth et Keane (1998) also point out the synergy between plan adaptation and retrieval. In our work, we take into account this synergy and we combine the similarity measure with other criteria to retrieve the most adaptable case.

According to Lopez de Mantaras et al. (Lopez de Mantaras et al, 2005), six types of retrieval related to the adaptation are identified: Diversity-Conscious Retrieval (Smyth and McClave, 2001 ; McSherry, 2002 ; McGinty and Smyth, 2003), Compromise-Driven Retrieval (McSherry, 2003, 2004), Order-Based Retrieval (Althoff and Bartsch-Spörl, 1996; Bridge and Ferguson, 2002) Explanation-oriented retrieval (Cunningham et al., 2003; Doyle et al, 2004), Optimization-Based Retrieval (Mougouie et Bergmann, 2002; Tartakovski et al, 2004) and Adaptation-Guided Retrieval (AGR). This present work is based on AGR.

3.2 Adaptation-Guided Retrieval phase.

The AGR assumes that the source cases most similar to the target case are not always the easiest to adapt, particularly when the similarity measure is based on surface characteristics. Retrieval therefore must not only search for similar cases but above all easily adaptable cases. Indeed, the authors explain how this retrieval which is guided by adaptation binds specifications and solutions spaces by using the knowledge adaptation. The work of Leake et al. (1997) addresses the adaptation effort concept and the impact of traditional semantic similarity measures on adaptation. Sometimes, the retrieved cases are “similar” to target problem but difficult or impossible to adapt. This generates an inability for the system to solve a problem or give a wrong answer squarely. Therefore, Leake takes into account the adaptation effort at the time of the retrieval step in order to facilitate the adaptation step. This consideration is concretized by inserting the “adaptation cost” in the similarity measure. Leake proposes two steps: first, a classic similarity measure is made by comparing case descriptors, then the most similar cases retrieved at the end of the first step are prioritized according to their adaptability.

To retrieve the best similar case for the adaptation in this work, two measures are implemented:

- the first one is a similarity measure (RM) taking into account all the descriptors informed. The retrieve phase associates the RM measure with a kNN algorithm in order to choose the set of the most similar cases to the target case
- The second one, is an “adaptation measure” (AM). It emphasizes on the components which exhibit an abnormal functional mode. The AM measure selects the most adaptable retrieved source case (observed case) from the similar cases selected by the RM measure.

3.2.1 Retrieval Measure

To retrieve the best similar case for the adaptation, first one needs to evaluate the similarity between descriptors and between the attributes of each descriptor. Indeed, for the

localization part, the problem descriptors of the target case and source cases will be compared. Then, concerning the functional part, it will be the attributes of descriptors which will be compared. Four local similarity measures are exploited.

In the first one, the hierarchical relation between the values of the descriptors is taken into account.

For the value of d_i^{value} which belongs to the hierarchical model of descriptors,

Let S_i be an expected state (source ds_i^{value} case) and T a component observed (target case).

$sim(d_i^{value}, d_i^{value})$ is developed by

$$\begin{aligned} sim(d_i^{value}(S_i), d_i^{value}(T)) &= sim(ds_i^{value}, dt_i^{value}) = 1 \quad \text{when} \quad ds_i^{value} = dt_i^{value} \\ &= 0,8 \quad \text{when} \quad level(ds_i^{value}) = level(dt_i^{value}) \\ &= 0,6 \quad \text{when} \quad 1\dots different \dots level \dots (ds_i^{value}, dt_i^{value}) \end{aligned}$$

For example: When the descriptor values are on the same level.

$$sim(ds_i^{value}, dt_i^{value}) = 0,8 \quad \text{if} \quad ds_i^{value} = D1 \quad \text{and} \quad dt_i^{value} = D2$$

When descriptor's values have parents with the same level.

$$sim(ds_i^{value}, dt_i^{value}) = 0,6 \quad \text{if} \quad ds_i^{value} = D1 \quad \text{and} \quad dt_i^{value} = Bal1$$

see Fig.4

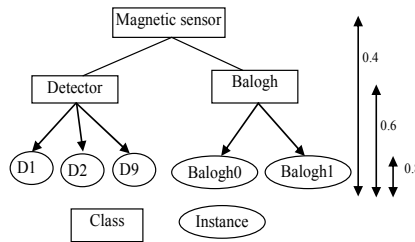


Fig. 4: Example of a descriptor hierarchy.

For the descriptor value ds_i^{state} and ds_i^{MF}

$$\begin{aligned} sim(ds_i^{state}, dt_i^{state}) &= 1 \quad \text{if} \quad ds_i^{state} = dt_i^{state} \\ &= 0 \quad \text{if} \quad ds_i^{state} \neq dt_i^{state} \end{aligned}$$

$$\begin{aligned} sim(ds_i^{MF}, dt_i^{MF}) &= 1 \quad \text{if} \quad ds_i^{MF} = dt_i^{MF} \\ &= 0 \quad \text{if} \quad ds_i^{MF} \neq dt_i^{MF} \end{aligned}$$

The similarity metric depends on the formalization of the case. Note that not all descriptors are filled. In order to compare cases in which the descriptors are not all filled, a similarity measure of the presence is defined. It will reflect the presence of descriptors in the case.

To take into account the presence and/or the absence of information in descriptors, a local similarity is developed.

$$\begin{aligned} sim_{presence}(ds_i^{value}, dt_i^{value}) &= 1 \quad \text{If information is present both in } S_i \text{ and } T \\ &= 0 \quad \text{if an information is not present in one descriptor} \end{aligned}$$

The global similarity measure (1) is obtained by aggregation of these functions on the whole set of descriptors. From this measure, a set of cases can be selected.

$$R_M(S_l, T) = \frac{\sum_{i=1}^p \text{sim}(ds_i^{\text{value}}, dt_i^{\text{value}}) + \sum_{i=p+1}^m \left[\text{sim}(ds_i^{\text{value}}, dt_i^{\text{value}}) \times \text{sim}(ds_i^{\text{state}}, dt_i^{\text{state}}) \times \text{sim}(ds_i^{\text{FM}}, dt_i^{\text{FM}}) \times \text{sim}_{\text{presence}}(ds_i^{\text{value}}, dt_i^{\text{value}}) \right]}{\sum_{i=1}^m \text{sim}_{\text{presence}}(ds_i^{\text{value}}, dt_i^{\text{value}})} \quad (1)$$

Where p represents the number of localization descriptor
m: represents the number of problem descriptors.

- Similarity Measure Calculation

$$R_M(S_4, T_1) = \frac{\dots d_1 \dots d_2 \dots d_3 \dots d_5 \dots d_7}{(0 \times 1) + (1 \times 1) + (1 \times 1 \times 1 \times 1) + (1 \times 1 \times 1 \times 1) + (1 \times 1 \times 1 \times 1)} = 0,8$$

$$R_M(S_9, T_1) = \frac{\dots d_1 \dots d_2 \dots d_3 \dots d_5 \dots d_6 \dots d_7}{(1 \times 1) + (1 \times 1) + (0,8 \times 1 \times 0 \times 1) + (1 \times 1 \times 1 \times 1) + (0,8 \times 1 \times 1 \times 1) + (1 \times 1 \times 1 \times 1)} = 0,8$$

3.2.2 Adaptation Measure

To determine the most adaptable case, an adaptation measure is established to take into account the functional modes associated to components involved in the case of diagnostic. Indeed, during a fault diagnostic, experts are particularly interested in cases with abnormal operating conditions. Therefore, the adaptation measure preferred descriptors associated with the functional mode and gives priority to abnormal functional modes.

The Adaptation Measure “AM” (2) takes into account the source cases supervisor descriptors (expected state) which are different from case target (observed component) and will be only linked to the class and to the functional mode compared to the solution descriptors. The adaptation measure is conditioned by the functional mode value. Indeed, a strong weight is affected to the dysfunctional mode related to the failure.

$$A_M(S_l, T) = \frac{\sum_{i=p+1}^m \left[\text{sim}(ds_i^{\text{value}}, dt_i^{\text{value}}) \times \lambda_i \right]}{\sum_{i=p+1}^m \text{sim}_{\text{presence}}(ds_i^{\text{value}}, dt_i^{\text{value}})} \quad (2)$$

Where p represents the number of localization descriptor

m-p-1: represents the number of supervisor descriptors

λ_i is the associated weight according to the functional mode.

- If $ds_i^{\text{MF}} = dt_i^{\text{MF}} = \text{nor} \rightarrow \lambda_i = 2^0$
- If $ds_i^{\text{MF}} \neq dt_i^{\text{MF}} \dots \dots \rightarrow \lambda_i = 2^1$ FM = normal $\rightarrow \lambda_i = 2^0 = 1$
- If $ds_i^{\text{MF}} = dt_i^{\text{MF}} = \text{ab} \dots \rightarrow \lambda_i = 2^2$

A weight is associated to the functional mode because the latter is considered as being important in the determination of the failing component. The number of different descriptors is determined by the denominator in equation (2). The retrieved source case having the

greatest adaptation measure value among the retrieval source cases will be the candidate chosen for the adaptation step.

- *Adaptation measure calculation*

The second step consists of applying the adaptation measurement (A_M) by taking the weight $\lambda_i = \{2^0, 2^1, 2^2\}$. This value considers that the abnormal functional mode is twice more important than the normal functional mode.

$$\begin{array}{l} \dots\dots\dots d_3 \dots\dots d_5 \dots\dots d_7 \\ A_M(S_4, T_1) = \frac{(1 \times 4) + (1 \times 1) + (1 \times 1)}{3} = 2 \end{array} \quad \begin{array}{l} \dots\dots\dots d_3 \dots\dots d_5 \dots\dots d_6 \dots\dots d_7 \\ A_M(S_9, T_1) = \frac{(0,8 \times 2) + (1 \times 1) + (0,8 \times 1) + (1 \times 1)}{4} = 1,1 \end{array}$$

We note that $A_M(S_4, T_1) > A_M(S_9, T_1)$. Thus, the case source 4 is the case which will be selected for the adaptation phase. We observe by at the same time that the source cases 4 and 9 are the same measures of similarity.

4 Adaptation Phase

Our objective in this study is to generalize the adaptation phase.

To obtain a general method in the field of diagnostic, we focused particularly on a unifying method. These methods have as objectives to propose general models from different angles (principles, algorithms, etc...). Hanney and Keane (Hanney and Keane, 1996) build adaptation rules from differences between attributes of cases pairs. Fuchs (Fuchs et al, 2000) defines the adaptation rules for interval calculations on dependent descriptors (The dependence of descriptors are determined by an expert.) Our adaptation method is inspired by Fuchs work which defines the dependency relations between problem and solution.

To set this dependency relationship between the symbolic data, we relied on the definition of the relevance between problem descriptor and specially the class descriptor.

If the variation of a descriptor d_i has an impact alone on the variation of D_j , we qualify the relation as high since d_i is strongly relevant with respect to D_j . A problem descriptor d_s is strongly relevant compared to a solution descriptor D_s when the value of d_s descriptor is crucial in the determination of D_s value. The change of d_s value is directly reflected on D_s value.

If this variation complete the variation of other descriptors, we have a weakly relevant with respect to D_j .

These relations identify for each couple (d_i, D_j =class) the existence of weakly relevant or strongly relevant features, this will select the kind of adaptation. Therefore three types of relationships are defined: [high, low, no relation].

Our data is symbolic; the adaptation rules must be detectable by using two models:

- context model that allows a set of components to adapt and,
- hierarchical components model that addresses the component class.

In this contribution, we propose an adaptation algorithm based on the context model, dependency relations between various problem and solution descriptors and descriptors

hierarchical model. If the solution class of the best chosen source case is similar to the problem class then the algorithm uses the hierarchical model. If the class is different, then the algorithm uses the contextual model to localize a set of potentially failing components and then uses the hierarchical model.

4.1 Dependency relations (DR)

4.1.1 Definition

The influence of a descriptor problem “ds” on the solution descriptors “Ds” is expressed by a dependency relation. A dependency relation is a triplet (ds_i, Ds_j, DR_{ij}) . DR_{ij} gives us the type of relationship between the problem and the solution to a given case. The relationship DR_{ij} takes its value in the set $DR_{ij} \subset (\text{No relation, Low, High})$.

- $DR_{ij} = \text{High}$: there is a high dependency relation between ds_i and Ds_j descriptors. Indeed, ds_i descriptor is strongly relevant compared to Ds_j descriptor.
- $DR_{ij} = \text{Low}$: there is a low dependency relation, i.e., the descriptors are connected thanks to the context which will be characterized by a contextual model.
- $DR_{ij} = \text{No relation}$: there is independency between ds_i and Ds_j .

These relations are based on the definition of relevance between descriptor problem and class of solution problem: The strong relevance, the low relevance and the non-relevance. The relevance notion is well known in the field of feature selection.

“A feature d_k is strongly relevant to sample N if there exist examples A and B in N that differ only in their assignment to d_k and have different labels (class) (or have different distributions of labels if they appear in N multiple times). Similarly, d_k is strongly relevant to target D_{class} and distribution D if there exist examples A and B having non-zero probability over D that differ only in their assignment to d_k and satisfy $D_{\text{class}}(A) \neq D_{\text{class}}(B)$ (Blum and Langley et al, 1997 definition’s pages 248-249)”.

“A feature d_k is weakly relevant to sample N (or to target D_{class} and distribution D) if it is possible to remove a subset of the features so that d_k becomes strongly relevant) (Blum and Langley et al, 1997 definition’s pages 248-249)”.

Remark

The measures AM and RM select the most adaptable case among the cases the most similar to the problem to be solved, while DR serves to qualify the type of the existing relations between the problem and the solution of the most adaptable case. This is done to select the category of adaptation.

4.1.2 Identification of DR.

There are three way to determine the relationship between the problem and the class.

1°- The expert determines this relation.

2°- The relevant descriptors leading to a class of failure can be identified thanks to the causal model and by the use of a fault tree which defines the relation between the symptom and the

cause. The causal model is easily built by using the tools recommended by Rasovska's methodology.

3°- The learning algorithm: To identify this type of relationship, we can apply a variables filtering algorithm such as Reporting Strong Algorithm for Subset Selection (STRASS) (Senoussi and Morello, 2008), which selects descriptors with high relevance and low relevance descriptors on a set of examples.

4.2 Adaptation Algorithm

The algorithm (algorithm 1) relies on the context model, the descriptors hierarchical model and the dependency relations. This algorithm checks descriptors one by one. The substitution's adaptation, by generalization and by specialization, will be taken into account in the algorithm.

Three possible scenarios are treated differently by the algorithm:

- DR = high and same class of problem and solution descriptors.
- DR = high and different class of problem and solution descriptors
- DR = Low

This algorithm deals with the adaptation of one descriptor at a time. It is conditioned by the solution descriptor class found at retrieval step. After the retrieval phase which makes it possible to select a retrieved case (ds_i^{ret1}, Ds_j^{ret2}) thanks to both RM and AM measures, the adaptation phase is launched. The initialization step creates a list of couples having a relation either high or low. According to the nature of the relation, the treatment differs. Consequently, the second step will depend on the DR values and the classes of the descriptors.

```

Input: retrieval case (dsi(ret), Dsj(ret))
Output: descriptors solution of the adapted case Dtj

For each Dsj(ret) do // j = 1..m.
    Create a list [DRij, Dsi] → Select the pair (DRij, Dsj(ret))
EndFor

For set of pairs (DRij, Dsj(ret)) do
    If (DR = high) then
        If Dsj(ret) and dsi(ret) are the same class then
            Dtj ← dsi(ret)
        ElseIf
            Select / "dt*" belonging to the same class as "remj Ds"
            Affect the solution value of the "dt*"
            Dtj ← solution of dt*
            GoTo EndFor
        EndIf
    EndIf
    If (DR = low) then
        Find the source problem descriptor associated with the problem
        target descriptor corresponding to the same class as the
        solution source descriptor substitution
        Substitute the new value of source solution descriptor by the
        corresponding value according to its class

        Dtj ← solution of dt*
    EndIf

    If (DR = none) then
        Nothing to make
    EndIf
EndFor

```

¹ Retrieval descriptors problem.

² Retrieval descriptors solution.

Algorithm 1. Adaptation algorithm

1. If by browsing through the list, a value of “DR = high” is found then the couple is selected and classes of “ Ds_j^{ret} ” and “ ds_i^{ret} ” are compared. If they have the same parent class, the influence of this substitution will be considered in “ Ds_j^{ret} ” and the algorithm will assign this new value to Dt_j^{ret} , otherwise, the algorithm looks at the context list descriptors and selects the “ dt_i ” descriptor which belongs to the same parent class as “ Ds_j^{ret} ”. The target descriptor is denoted “ dt^* ”. Then, the value of the reminders will be determined and will be thereafter affected to “ Dtj ”.
2. If in the list there is only DR = low then the algorithm selects the parent class of Ds_j^{ret} descriptor. Then, it identifies the dt_i descriptor belonging to the same parent class as Ds_j^{ret} which will change status ($dt_i \rightarrow dt^*$). After that, the relationship dt^* will influence the transformation of the Ds_j^{ret} solution which will be affected thereafter to “ Dtj ”.
3. Finally, when all DR values are equal to “no relation” then there is no adaptation.

5 Retrieval and Adaptation Illustration

The diagnostic cases modeling took a specific form, described in Table 3. The three kind of adaptation which will be approached are illustrated in this Section.

Kind 1: DR = High and same functional class

Suppose a failure occurred in the D9 detector. The retrieved source cases closest to this target case 1 (noted “ T_1 ”) provide two sources cases: the source case 4 S_4 and the source case 9 S_9 (see Table 3).

Table 3: 3 kinds of adaptation.

Kind 1	d1	d2	d3			d4			d5			d6			d7		
T1	Puller Zone	exit	D9	1	an				STP6	top	nor	D9	0	nor	Bal1	1	nor
S4	Internal Ring	exit	D8	1	an	Puller	down	nor	STP6	top	nor				Bal1	1	nor
S9	Puller Zone	exit	D6	1	nor				STP6	top	nor	D8	0	nor	Bal1	1	nor
Kind 2	d1	d2	d3			d4			d5			d6			d7		
T2	Post Zone	Ind	D1	1	nor	Pusher	top	an	STP4	top	nor	D5	0	an	Bal0	1	nor
S3	Post Zone		Bal1	1	nor	Ind	top	an	STP4	top	nor						
S6	Post Zone	Ind							STP4	bot	an	D6	0	nor			
S8	Robot Zone	Ind	Bal0	1	nor	Ind	top	an	STP4	top	nor	D6	0	nor			
Kind 3	d1	d2	d3			d4			d5			d6			d7		
T3	External Ring	Ext conv	D4	0	nor				STP3	bot	nor	D5	0	nor	Bal1	1	nor
S5	External Ring	Ext conv	D6	1	nor				STP5	top	nor	D5	0	nor	Bal1	1	nor
S7	External Ring	Ext conv	Bal1	0	nor	Pusher	down	nor	STP5	top	an	D7	0	nor	Bal1	1	nor

The applied similarity threshold is 60%.

Similarity Measure R_M and adaptation measure Calculation A_M are shown in Table 4.

We observe at the same time that the similar source cases are S4 and S9 and they have the similar value of R_M , but differ in the A_M value

It is important to note that the DR value of the pair (Ds_2 , ds_3) is “DR = high” and that the D8 detector which is shifted of the Ds_2 descriptor as well as the D9 detector of the descriptor dt_3 belong to the same class “presence sensor” (Table 4 bis).

Table 4: AM and RM calculation of case 1.

	T1 S4	T1 S9
RM	$(0+1+0,8*1+1+1)/5 = 0,76$	$(1+1+0+1+0,8+1)/6 = 0,8$
AM	$(0,8*4+1+1)/3=1,73$	$(0,8*2+1+0,8*1+1)/4=1,1$

Table 4bis: Context of the case.

case 4	Ds_2	Identification of the failure component			
	ds_i	ds_3	ds_4	ds_5	ds_7
subst. ds_3 by dt_3	value S4	D8	Puller	STP6	Bal1
	value T1	D9		STP6	Bal1
	$DR(Ds_2; ds_i)$	high	low	low	low
Result=D9	DS_1 (Class)	Presence sensor	Actionner pneumatic	Electrical actuator	magnetic sensor
	DS_2 (T1)	D9	DT_1 : (Class)	Presence sensor	

Consequently:

- Substitution of the ds_3^{ret} by “ $dt_3 = D9$ ”.
- The new value of ds_3^{ret} will affect the value of DS_2^{ret} which will get the value $DS_2^{ret} =$ shifted D8.
- Assignment of the new value of DS_2^{ret} to the target solution Dt_2 .

By applying the adaptation algorithm the solution is as follows:

The failed component is: ***the detector D9, which is located in the entry of the external ring***

The repair action is: ***Replacement of the failed component.***

Kind 2: DR = High and different functional class

Suppose a failure occurred in the “post zone” and represented by target case 2, denoted “T2” in Table 5. The results of the calculation of retrieval and adaptation measures are the following:

According to the results of the adaptation measure(s), the selected source case for the adaptation phase is case 6, **S6**.

The DRs values of the source case 6 are as follows :(Table 5bis)

Ds4: Indexer of the post zone = Ft(ds₁: post zone, ds₂: indexer).

Ds2: Balogh1 (strong magnetic fields) = Ft(ds₅: STP4; DR₅₂ = high, ds₆: D6; DR₆₂ = low).

We note that the DR value of pair (Ds₂, ds₆) is “DR = High” and “D6” and “Balogh1” components do not belong to the same family. This confirms that we are in the scenario of: “DR = High and different functional classes”.

Table 5: AM and RM calculation of case

	T2 S3	T2 S6	T2 S8
RM	$1+0+0,6+0,8+1)/4 = \mathbf{0,85}$	$(1+1+0+0,8*0)/4 = \mathbf{0,5}$	$(1+1+0,6+0,8+1+0,8*0)/6 = \mathbf{0,733}$
AM	$(0,6+4*0,8+1)/3 = \mathbf{1,6}$	$(1+0,8*4)/2 = \mathbf{2,1}$	$(0,8+4*0,8+1+2*0,8)/4 = \mathbf{1,65}$

Table 5bis: Context of the case.

case 6	Ds ₂	Identification of the failure component	
		ds ₅	ds ₆
	descriptor	ds ₅	ds ₆
subst. ds3 by dt3	value S6	SPT4	D6
	value T2	SPT4	D5
	DR	low	high
Result= Bal0	D ₁ (S6)	Electrical actuator	Presence sensor
D ₁ (T1):	magnetic detector	DS ₂ : T1	Bal0

By applying the adaptation algorithm, we obtain:

- The class of source solution descriptor “ ds_2^{ret} ” is “Magnetic detector”
- The “STP4” component of the target descriptor “dc5” (which corresponds to the descriptor “ds5” which is in abnormal mode) is in the context of the “Indexer of the post zone”. However, in this zone there are other components: “D1, pusher, D5 and Bal0”
- The “Bal0” component belongs to the same class of Dsret2 which is “Magnetic detector”
- Substitute the “Bal1” value of the “ ds_2^{ret} ” descriptor by the “Bal0” value of “dc7”.

Thus, $ds_2^{ret} = \text{Bal0}$ (strong magnetic fields)

- Assign this value to “Dc₂”: Dc₂ = Bal0 (strong magnetic fields)

The solution is as follows:

By applying the adaptation algorithm the solution is as follows:

The failed component is: *The Balogh: Bal0, belonging to the “Magnetic detector” class, which is located at the indexer of the post zone.*

The repair action is: *Clean the Balogh: Bal0.*

Kind 3: DR = Low

Suppose a failure occurred in the “external ring” and represented by target case 3, denoted by T3 in the Table 6. The results of the calculation of retrieval and adaptation measures are the following:

Table 6: AM and RM calculation of case 3.

	T3 S5	T3 S7
RM	$(1+1+0+0+1+1)/6=0,667$	$(1+1+0,6+0,8*0+0,8+1)/6=0,733$
AM	$(0,8+0,8+1+1)/4=0,9$	$(0,6+0,8*2+0,8+1)/4=1$

Table6bis: Context of the case

	Ds ₂	Identification of the failure component			
case 7	descriptor	ds ₃	ds ₅	ds ₆	ds ₇
contextual	value S7	Bal1	STP5	D7	Bal1
	value T3	D4	STP3	D5	Bal1
	DR(Ds ₂ ;ds _i)	low	low	low	low
subst. ds ₃ by dt ₃	D ₁ (S7)	magnetic detector	Electrical Actuator	Presence sensor	Magnetic Sensor
Result=Pusher	D ₂ (T3): dt ₃	Pusher		D ₁ (T3)	Pneumatic Actuator

The DRs values of the source case 7 are as follows (Table 6 bis)

Ds₄: External conveyor of exit external ring = Ft (ds₁: exit of external ring, ds₂: external conveyor).

Ds₂: Blocked puller = Ft(ds₃: Bal1; DR₃₂ = low, ds₅: STP5; DR₅₂ = low, ds₆: D7; DR₆₂ = low, ds₇: Bal1; DR₇₂ = low).

We note that all couples (Ds₂, ds₃), (Ds₂, ds₅), (Ds₂, ds₆) and (Ds₂, ds₇) have values of “DR = Low”

By applying the adaptation algorithm, we obtain the following steps:

- The solution descriptor “D₅^{ret}” is from “pneumatic actuator” class.
- The context of the target case components “D4, STP3, D5 and Bal1” is located at the external conveyor of the secondary ring.

- In this context, the component pertaining to the same context and which belongs to the same class of Ds_2^{ret} (pneumatic actuator) is the component: “pusher”.
- Substitution of the “puller” value by “pusher” value in the descriptor Ds_2^{ret} , $Ds_2^{ret} =$ blocked pusher.
- Assign this new value to the “Dc2” descriptor.

By applying the adaptation algorithm the solution is as follows:

The failed component is: ***The pusher in the external conveyor of the exit external ring***

The repair action is: ***Unblock the pusher.***

Furthermore, we have found through these three examples that the source case which is the most similar to the target is not necessarily the case chosen for the adaptation phase

We applied this method for a car system with a diesel motor 1.5 dCi K9K 105ch of from Renault (Chebel-Morello et al 2009). In order to validate the results obtained by using our method and to compare it with a traditional one that uses simple retrieval and adaptation phases, we will conduct a validation method. This method will be described in the next section.

6 Validation & Discussion

In this section we present three experiments concerning our case based reasoning system:

The first one concerns the need for the adaptation phase in our system. Accuracy of the diagnostics system with and without adaptation phases is compared.

The second one compares the computational speed of the system in the adaptation guided retrieval AGR phase with the retrieval classical phase.

The second one evaluates the computational speed of diagnosis reasoning (retrieve+ reuse phases) and compares it in the adaptation guided retrieval AGR phase with the classical retrieval phase.

And the third one studies the performance of the adaptation algorithm.

The main goal of these experiments is to prove the feasibility of the proposed method of diagnostics composed of a case base, knowledge model, AGR and adaptation method. We used a leave-one-out cross-validation method for the first two parts to assess SISTRE's ability to accurately adapt retrieved cases for a case base containing 125 cases.

6.1 Accuracy of the adaptation phase

In this part, the accuracy rate with and without adaptation will be calculated. The results show that the proposed method with the adaptation selects the cases which are the best adaptable ones and thus it gives an accuracy rate of 88% (See **Fig.5** and **Fig.6**). If the adaptation algorithm is powerful, one can get a good performance concerning the CBR system applied to a limited number of cases.

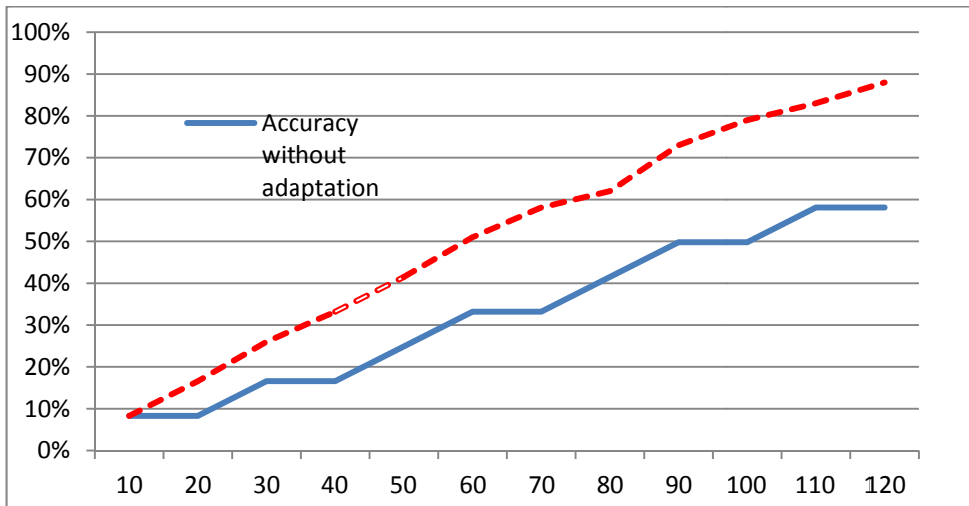


Fig. 5: Evolution of Accuracy following the number of cases in the test-base

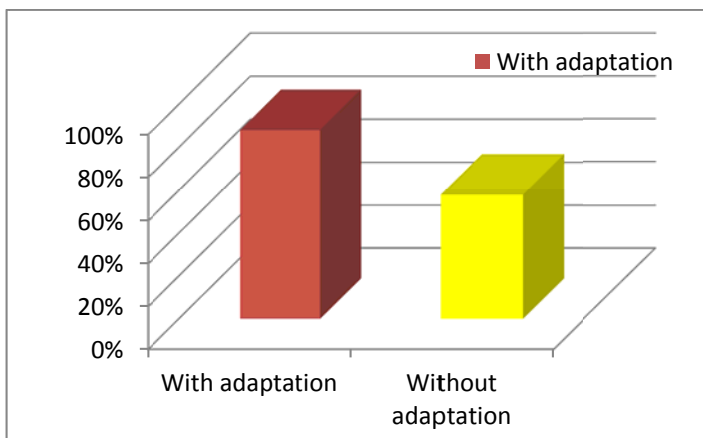


Fig. 6: Accuracy with and without adaptation

However, we find bad results without the adaptation, an accuracy rate of only 58.1% with the retrieval step. These results show that the case base is not complete and gives without adaptation a bad recognition rate. The adaptation phase and models of knowledge on which it relies are essential. Our diagnostic system gives a good recognition rate.

6.2 AGR method

This experiment is designed

To evaluate the solving time of the help diagnostics system which use knowledge model and to compare it to the help diagnostics system with two types of the retrieval phase.

We start with the comparative study of our adaptation-guided retrieval method. Initially, we apply the adaptation algorithm after a retrieval step exploiting the two RM and AM measures. Then we apply the algorithm after a retrieval stage exploiting only the similarity measure RM.

Fig. 7 shows the obtained results compared to the computational speed of the adaptation algorithm in 10^{-3} seconds. The adaptation algorithm is executed on a computer with a microprocessor Intel (R) Core (TM)2 Duo CPU “2.66GHz”.

The algorithm of adaptation, which is in relation to the knowledge of the system SISTRE through the models of knowledge, reacts differently depending on the AGR principle.

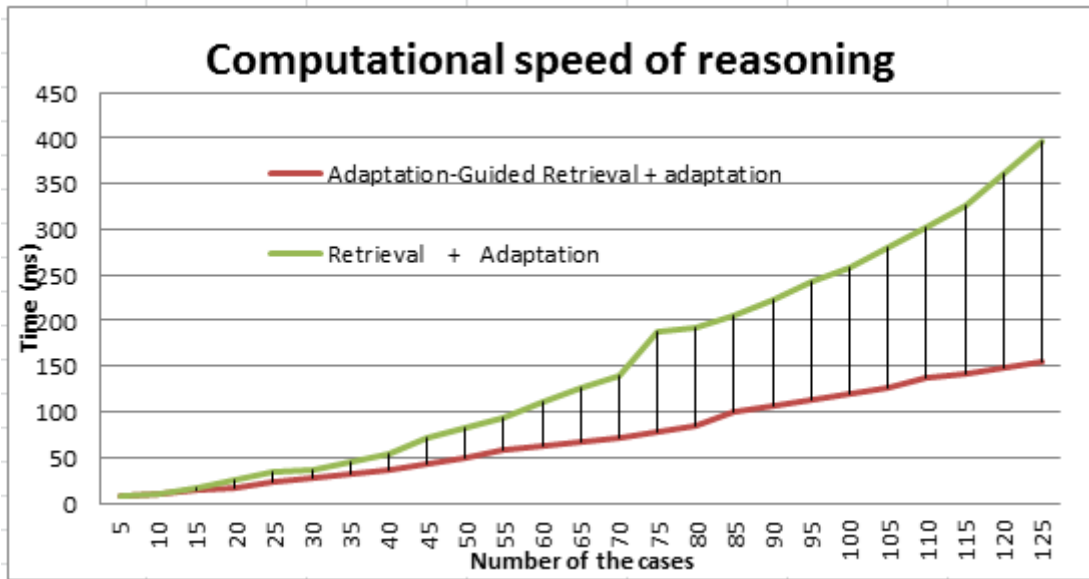


Fig.7: Analysis of the adaptation-guided retrieval method

According to **Fig.7** which shows the elapsed adaptation time in ms, we note that the adaptation effort is less important when the AM adaptation measure is employed (in the AGR case). In fact, the larger the case base it, the more important the variation of the adaptation effort is. Consequently, the adaptation measure which is resulting from the AGR principle decreases considerably the adaptation effort. Moreover, this principle allows selecting the good case that will lead easily and correctly to the given final solution according to the equipment knowledge context. Moreover, retrieval using only the similarity measure without adaptation does not select the cases which are the best adaptable ones (in particular concerning their classes). These results also show that the most adaptable is not necessarily the most similar.

6.3 Adaptation phase

This experiment is designed to study the accuracy of the help diagnostics system; overall accuracy, and more precisely the accuracy of only retrieval cases. The results of this study are shown in Table 7

Table 7: Results of the adaptation.

Total cases	Successful adaptation	Failed adaptation	Failed retrieval	Overall accuracy	Accuracy of only retrieved cases
125	110	10	5	88%	91.66%

We note that the accuracy rate 88% reflects that 110 cases were adapted correctly to the set of 125 cases. This accuracy is computed using “Ds₂” as the component responsible for

the failure. By analysing the results, one can find that there are 5 retrieval failure cases. These cases are considered as generic cases that are covered by no other case of the case-base. Thus, their adaptation is impossible.

To evaluate the performance of the adaptation algorithm, these 5 cases are not considered in the accuracy calculation. For this subset of cases the accuracy rate is 91.66%. These algorithms can be applied to any type of discrete event systems (transfer systems). To prove the genericity of this method on any type of industrial plant, we applied these algorithms to another type of equipment, a diesel engine, but on a limited number of cases (Haouchine, 2009).

6.4 Comparison with other in industrial diagnosis system.

6.4.1 State of art in industrial diagnostic.

Concerning the reasoning process in the industrial diagnosis system, Gas Turbine, IRACUS and NodalCBR systems do not use adaptation: they offer directly the solution of the retrieved case. These systems are not effective because the retrieved case is not necessarily the chosen case for adaptation. As for Cassiopee, Pad'im (Mille et al, 1999) and Patdex systems, they propose simple adaptation rules and cannot ensure an accurate solutions' result of the adapted cases. FormTool (Cheetham et Graf, 1997) exploits the relation between the retrieval and the adaptation phases. However, even if FormTool relies on this relation, the technique used is strictly dedicated to the treatment of plastic colors following a specific function. Consequently, it is not easily exploitable in other fields and cannot adapt other cases outside the characteristics of plastic colors.

Finally, the Creek system is the most effective among them because it proposes an explanation oriented retrieval and it exploits more complex and more complete relations in its adaptation phase thanks to models. In the retrieval step, the Creek system uses a similarity measure combined with measures related to the computation path in a semantic network. Concerning the adaptation step, it does not give any algorithm and it does not show how this step is carried out. The adaptation begins by activating a set of cases that are matches, and the step that refines the selection of a subset of cases by exploiting the non-identical explanation characteristics of each case. Lastly, the final focusing step is based on the solution and that's by selecting the best case corresponding to the request and by copying or modifying the solution of this case. The adaptation phase is complex and is usually designed for a specific application.

6.4.2 Comparison methodology

To compare our diagnostic method to other CBR diagnostic method, we define three indicators linked to the elicitation and reusing knowledge in the new application.

CBR systems differ from each other in the development tool, the formalization of the case, the use of knowledge models, the different phases of the CBR cycle and the different algorithms and methods used in each step. Table 8 summarizes these points associated with the effort of creating the knowledge models the cost of adaptation and the cost of re-use.

Table 8. Comparison of the CBR diagnostic system.

CBR Systems	creation effort	development tool	system oriented	Retrieval	adaptability cost	Adaptation	Reuse cost	Knowledge	Global cost
Gas Turbine	8	Tool circel	knowledge	normal	10	No adaptation	10	Bayesian network	28
IRACUS	10		classification	normal	10	No adaptation	10	no model	30
Nodal _{CBR}	8	ICEE	knowledge		10	No adaptation	10	classification tree	28
Cassiopee	8	Kate Kaidara	knowledge	Normal	5	Transformati onnel	5	conversationnel	18
Patdex	5		knowledge	Normal	5	Graphs of experiments	5		15
FormTool	10		classification	AGR	0	Specific function	10	no model	20
Creek	2		knowledge	AGR	0	Trace of reasoning	7	semantic network	9
Caseline	8	Kate Kaidara	knowledge	Normal	10	Transformati onnel	5	no model	23
Our method	2		knowledge	AGR	0	method	3		5

1°- **The effort of creating the knowledge models** can be quantified by a development cost.

To use development systems one needs to buy the system, to understand it and develop models. This requires expert availability. We will identify a cost of 8 on a scale of 0 to 10. However, representing expert knowledge from scratch will cost 10.

The creation of the knowledge model without the development system relies on the analysis of functional and dysfunctional models like Creek and our method requires less effort to model the creation. We can estimates the cost of this complex model to be 3

2°- To estimate the capacity to reuse the cases in the retrieval phase, we define the easiness of the adaptation by a **cost of adaptation**.

Adaptation guided retrieval method takes into accounts the cost of adaptation to select the most suitable course. The adaptation cost will be minimized (= 0) in this case. When there is an adaptation in the CBR system, the cost of adaptation is equal to 5

When there is no adaptation in the CBR system, the cost of adaptation is equal to 10.

Creek system is the most effective among the other diagnostic methods because it proposes an explanation oriented retrieval and it exploits more complex and more complete relations in its adaptation phase thanks to models.

3 °-For the re-use of the program of adaptation in the new application, **the cost of re-use** will depend on the generality of Adaptation methods.

If there is no algorithm of adaptation or there is a specific algorithm of adaptation the cost is maximal because a new algorithm must be developed. The cost of re-use is maximal, it is equal to 10.

If the application takes support on specific rules or models of knowledge the cost can be averaged, and it is equal to 5

When a method is independent of the application, as it is the case for our method, the cost is minimal and it is equal to 0.

Applying these criteria to the methods we obtain Table 8:

We can see that our method was implemented to minimize the cost of designing a knowledge system and therefore it gives better results than other methods.

7 Conclusion

A knowledge oriented diagnostics system integrated in the e-maintenance platform is proposed in this paper. The case based reasoning system is well adapted to solve fault diagnostics issues because problems are recurrent and cases can be reused. A state of the art in the frame of case based reasoning for technical diagnostics highlighted that there is no common method to build a diagnostic support system. To conceive this system, we have followed the method of Rasovska which is to develop the two processes of Lamontagne model (offline and online processes):

1. Firstly we developed the following knowledge containers: (i) vocabulary that takes into account the definition of diagnostics.(ii)case base that formalizes the experience;(iii)two similarity metrics (one for the retrieval phase and one for the adaptation phase) and knowledge adaptation with two models.

We set up a formalization of the cases and we associated to the descriptors a hierarchical model. This model is common to problem and solution descriptors of the case-base cases and a model relating to the application context. All steps depend on the cases formalization and the associated knowledge models. The elaboration of this expert knowledge has a cost, but this cost is minimized because the first model which is the component taxonomy model is well known by the maintenance users. However, the second model requires a minimum of work. This modeling has influenced the proposed similarity measure as well as the adaptation measure. The latter is directly related to the functional mode of the supervised components (an attribute specific to the descriptor).

2. Secondly, concerning the reasoning process, an adaptation-guided retrieval method has been proposed.

The retrieval phase is related to the adaptation phase using the conjunction of similarity and adaptation measures. This conjunction makes it possible to select among the retrieved cases the most adaptable. The adaptation phase will exploit the dependency relations between the problem and the solution. These dependency relations will be given either by the selection of a relevant descriptor or by the use of a context model between the various failures which can appear in an industrial plant. And the results depend on this relation. One difficulty to overcome in this method is to define these relationships in the case base.

The adaptation algorithm detects three scenarios and proposes associated actions to each case. We detailed one scenario where the most similar case is not necessarily the one selected for the adaptation.

Our previous studies have enabled us to formalize the case of a supervised industrial system of pallets transfer (SISTRE) in this paper and in a diesel engine, but on a limited number of cases. The proposed adaptation-guided retrieval and adaptation methods for a diagnostic application have been validated on a set of 125 cases resulting from the study carried out on SISTRE.

We have proved the feasibility of this diagnostic support system. To build it in any type of industrial equipment, two knowledge models need to be elaborated. To avoid the cost of the development of knowledge models, we are currently working on the use these algorithms with models (functional events and components models) developed in a web-maintenance platform. This model is defined in the domain ontology of maintenance, in the context of Semantic-maintenance and life cycle (SMAC) Project.

This method needs to be applied on a large scale that would attest its applicability for any type of equipment. We will apply this method to more complex equipments, where we select different levels of granularity in the equipment, and this will affect the formalization of the case.

ACKNOWLEDGMENT

This work was carried out and funded in the framework of SMAC project (Semantic-maintenance and life cycle), supported by European program Interreg IV between France and Switzerland.

APPENDIX A: THE SUPERVISED INDUSTRIAL SYSTEM FOR PALLET TO DIAGNOSTIC.

To illustrate each notion of this study, the industrial application to be developed is described in this section. This application is a supervised industrial system for pallets transfer (SISTRE). A description of the system is given in Fig. 8a and Fig.8b.

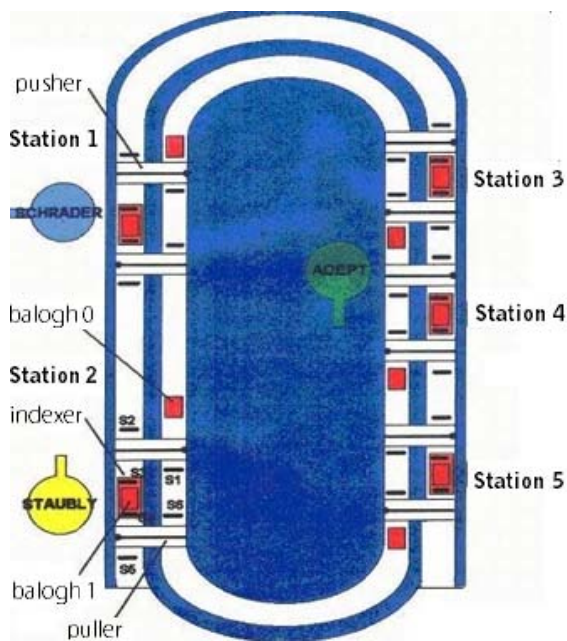


Fig. 8a: SISTRE.

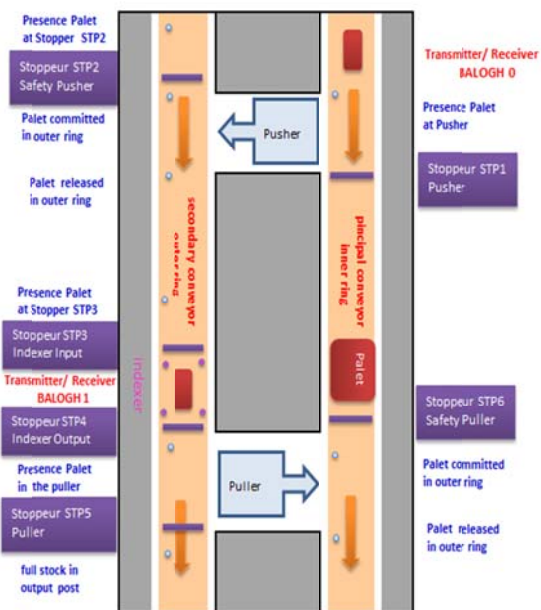


Fig. 8b: The working station of SISTRE.

The latter represents a flexible production system, composed of five robotized working stations which are served by a transfer system of pallets organized into double rings (internal and external).

As shown in Fig 8b, each station is equipped with pneumatic actuators (pushers, pullers and indexers) and electric actuators (stopper) as well as a certain number of inductive sensors (proximity sensors). An inductive read/write module allows to identify and locate each pallet and to provide information relative to the required operation in a concrete station. The motion of the pallets is ensured by the friction on belts that are powered by electric motors. Each pallet has a magnetic label that is used like an embedded memory. This memory can be read in each working station thanks to magnetic read/write modules (Balogh) and allows the memorization of the product assembly sequence. These labels thus enable to track the pallet

path through the system. The feasibility of our approach was studied in Section 6 through 125 generic cases q.

Functional analysis of the system has identified main, additional (compelmentary) and secondary functions. Main functions are as follows: transfer assembly parts between stations; identify pallet and product lines; store product information; transform Content of the pallet. Secondary functions secure the transfer system; provide pneumatic energy and electrical energy. Secondary functions are concerned with adding/removing pieces of pallets, and managing assembly lines.

The base case is studied in relation to the functional analysis of the level 2 on a station which is shown in Figure 9.

The studied case base is in relation with the internal functional analysis of level 2 relative to the working station that is represented in figure 9.

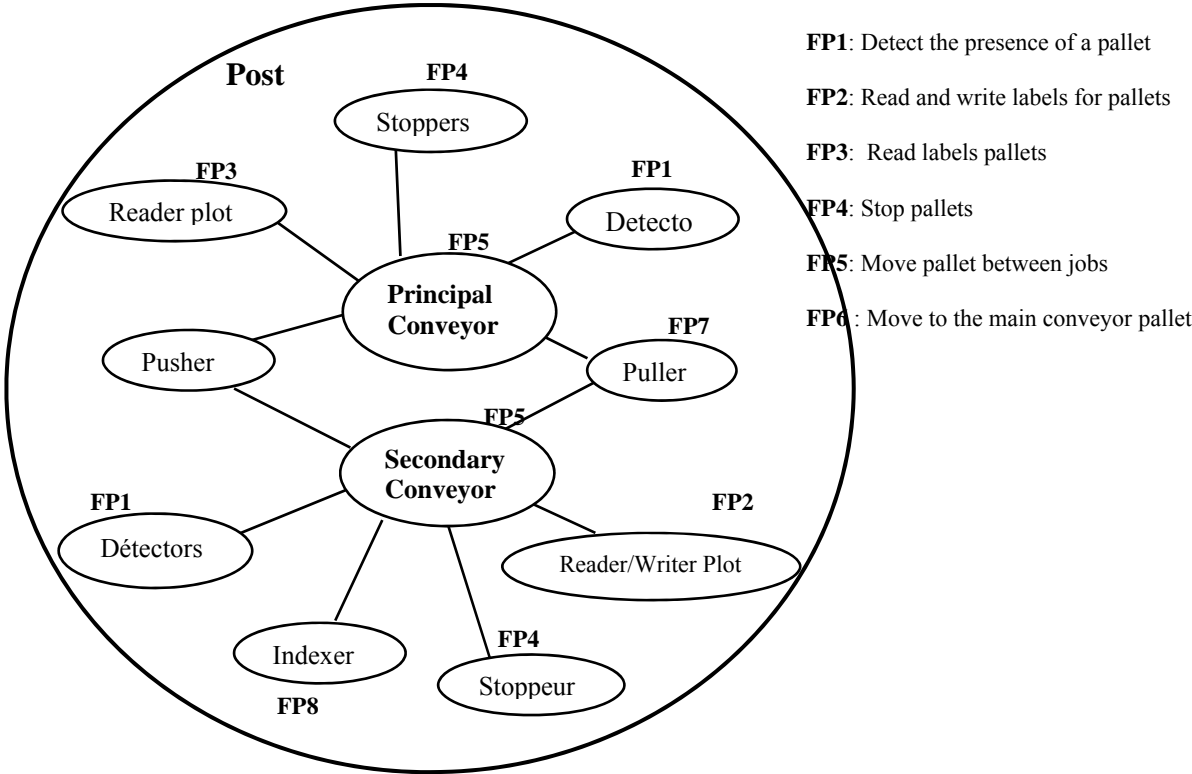


Fig. 9: Functional Analysis Internal Level 2 (Post)

REFERENCES

Aamodt, A, 2004. Knowledge-Intensive Case-Based Reasoning and Sustained Learn Proc. of the 9th European Conference on Artificial Intelligence, ECCBR04, Lecture N in Artificial Intelligence, pp.1-15, Springer, (2004)
 Althoff K.D. et Bartsch-Spörl B., 1996, Decision support for case based applica Wirtschaftsinformatik, ISSN 0937-6429, 1996, vol. 38, no1 pp. 8-16.
 Bergmann R., Althoff K.D., Breen S., Gker M., Manago M., Traphner R., Weiss S., 2003: Developing Industrial Case-Based Reasoning Applications: The INRECA Methodology. Lecture Notes in Artificial Intelligence, LNAI 1612, Springer Verlag, Berlin.

- BLUM A. L. and LANGLEY P. 1997, "Selection of relevant features and examples in machine learning", *Artificial Intelligence* 97(1-2), 1997. pp. 245-271.
- Bridge, D., Ferguson, A., 2002, An expressive query language for product recommendation systems. *Artificial Intelligence Review* 18(3-4), 269-307 (2002)
- Chebel-Morello B., Haouchine M.-K., Zerhouni N., 2009. A methodology to conceive a case based system of industrial diagnosis. In *World Congress of Engineering Asset Management, WCEAM'09*, Greece
- Chebel-Morello B., Haouchine M.-K., Zerhouni N., 2011 Adaptation based on knowledge models for diagnostic systems using case-based reasoning KMIS 2011 du 26 octobre au 29 octobre 2011 à Paris.
- Cheetham, W., Graf, J., 1997, Case-Based Reasoning in Color Matching, *Proceedings of Second International Conference on Case-Based Reasoning Research and Development Lecture Notes In Computer Science*; pp.1-12, vol. 1266, (1997)
- Cordier, A., Fuchs, B., Mille, 2006, A.: Engineering and Learning of Adaptive Knowledge in Case-Based Reasoning. 15th International Conference on Knowledge Engineering and Knowledge Management EKAW'06, S. Staab and V. Svatek Podesbrady, Czech Republic, 303-317, LNAI 4248. Springer-Verlag Berlin Heidelberg
- Cordier A., 2008, Interactive and Opportunistic Knowledge Acquisition in Case-Based Reasoning, PhD thesis, Laboratoire d'Informatique en Images et Systèmes d'information University of Lyon I, November.
- Cunningham P., Doyle D. et Loughrey J., 2003 An evaluation of the usefulness of case based explanation. In *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Berlin: Springer, pp. 122-130, 2003.
- Devaney, M., Cheetham, B., 2005: Case-Based Reasoning for Gas Turbine Diagnostic 18th International FLAIRS Conference (FLAIRS-05).
- Doyle, D., Cunningham, P., Bridge, D., Rahman, Y., 2004. Explanation oriented retrieval. In *Proceedings of the Seventh European Conference on Case-Based Reasoning*. Berlin: Springer, pp. 157-168
- Fuchs B., Lieber, J., Mille, A., Napoli, A., 2000: An Algorithm for Adaptation in Case-Based Reasoning. In the 14th European Conference on Artificial Intelligence (ECAI00), Amsterdam, The Netherlands, August 20-25, IOS Press, volume 14, pp 45-49,
- Grant P.W., Harris P.M. et Moseley L.G. 1996, Fault Diagnosis for Industrial Printers Using Case-Based Reasoning. *Engineering Applications of Artificial Intelligence*. Vol.9, No.2, pp.163-173.
- Grundstein, M. 2000, From capitalizing on Company Knowledge to Knowledge Management, In D. Morey, M. Maybury, & B. Thuraisingham (Eds), *Knowledge Management, Classic and Contemporary Works* (chapter 12, pp. 261-287). Cambridge, Massachusetts: The MIT Press,
- Hanney K., Keane M.T., 1996: Learning Adaptation Rules from a Case-Base. In *Proceedings of the 3rd European Workshop on Advances in Case-Based Reasoning*, Lecture Notes In Computer Science.
- Haouchine, M.K., Chebel-Morello, B., Zerhouni, N., 2008: Adaptation-Guided Retrieval for a Diagnostic and Repair Help System Dedicated to a Pallets Transfer. In *3rd European Workshop on Case-Based Reasoning and Context-Awareness*. 9th European Conference on Case-Based Reasoning, ECCBR 2008, Trier, Germany
- Haouchine, M.K. 2009 Rememoration guide par l'adaptation et maintenance des systèmes de diagnostic industriel par l'approche du raisonnement partir de cas. PhD thesis, Automatic en Micro-Mecatronic Department, Franche-Comte University,
- Jacobson, A., Prusak, L. 1996 The Cost of Knowledge 13th National Conference on *Artificial Intelligence*, AAAI Press, Stanford, California (1996).

- Karray H., Chebel-Morello B., Zerhouni N. 2009 «Towards a maintenance semantic architecture» World Congress of Engineering Asset Management, WCEAM'09., Grèce, (sept 2009)
- Karray H., Chebel-Morello B., Zerhouni N. 2011 A contextual semantic mediator for a distributed cooperative maintenance platform. 8th IEEE International Conference on Industrial Informatics 8th IEEE International Conference on Industrial Informatics, INDIN'10 Japon
- Kasif, S., Salzberg, S., Waltz, D., Rachlin, J., Aha, D., 1995: Towards a Framework for Memory-Based R. NECI Technical Report.
- Lamontagne, L., Lapalme, G. 2002: Raisonnement à base de cas textuels – état de l'art et perspectives, *Artificial Intelligence Review*, Hermes, Paris, vol. 16, no. 3, pp. 339-366
- Leake, D.B., Kinley, A., Wilson, D., 1997 CB Similarity Assessment: Estimating Adaptability from Experience. In Fourteenth National Conference on Artificial Intelligence, Menlo Park, CA. AAAI Press, pp. 674-679.
- Lieber, J., 2007: Application of the Revision Theory to Adaptation in Case-Based Reasoning: the Conservative Adaptation. In 7th International Conference on Case-Based Reasoning - ICCBR'07, 4626, pp. 239-253.
- Lopez de Mantaras R., McSherry D., Bridge D., Leake D., Smyth B., Craw S., Faltings B., Maher M.L., Cox M., Forbus K., Keane M., Aamodt A., Watson I., 2005 : Retrieval, Reuse, Revise, and Retention in CBR. *Knowledge Engineering Review*, pp. 215-240.
- Maintenance terminology. 2001 European standard, NF EN 13306.
- McGinty, L., Smyth, B., 2003: On the role of diversity in conversational recommender systems. *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Berlin: Springer, pp. 276-290.
- McSherry, D., 2004 : Balancing user satisfaction and cognitive load in coverage-optimised retrieval. *Knowledge-Based Systems* 17, 113-119.
- McSherry, D., 2003: Similarity and compromise. In *Proceedings of the Fifth International Conference on Case-Based Reasoning*. Berlin: Springer, pp. 291-305 (2003)
- McSherry, D., 2002 : Diversity-conscious retrieval. In *Proceedings of the Sixth European Conference on Case-Based Reasoning*. Berlin: Springer, pp. 219-233
- Mille. A. Fuchs, B. Chiron B. (1999) Reasoning based on the experiment: a new paradigm in industrial supervision « in french ». *Revue d'intelligence artificielle*, 13:97-128
- Mougouie, B., Bergmann, R, 2002: Similarity assessment for generalized cases by optimization methods. In *Proceedings of the Sixth European Conference on Case-Based Reasoning*. Berlin: Springer, pp. 249-263.
- Rasovska, I., Chebel-Morello, B., Zerhouni, N., 2008 A mix method of knowledge capitalization in maintenance in *Journal of Intelligent Manufacturing* Springer Netherlands ISSN0956-5515 (Print) Volume 19, Number 3 / juin 2008 pp 347-359
- Richter M.M., Wess S., 1991 Similarity, Uncertainty and Case- Based Reasoning in PATDEX. *Automated reasoning, essays in honour of Woody Bledsoe*, Kluwer, pp. 249-265.
- Richter M.M., The Knowledge Contained in Similarity Measures. Invited talk, First International Conference on Case-Based Reasoning (ICCBR'95)
- Riesbeck, C.K., Schank, R.C, 1989 : *Inside Case-Based Reasoning*. Hillsdale, New Jersey: Lawrence Erlbaum Associates, Inc, Cambridge, MA.
- Senoussi H. et Chebel-Morello M., (2008) A New Contextual Based Feature

- Selection. *IEEE World Congress on Computational Intelligence*, CDROM (WCCI 2008) Hong Kong, June 1-6 2008
- WCCI'08., Hong Kong : Chine, 2008. Smyth, B., Keane, M.T, 1993.: Retrieving Adaptable Cases. The role of adaptation knowledge in case retrieval, In the European Workshop on Case-Based Reasoning (EWCBR93), pp. 209-220, Kaiserslautern, Germany, Springer.
- Smyth, B., Keane, M.T 1995.: Experiments on Adaptation-Guided Retrieval in Case-Based Design. In: Veloso, M., Aamodt, A.; (eds.): Proceedings of the 1st International Conference on Case-Based Reasoning. LNAI, Vol. 1010, Springer, Berlin 313-324
- Smyth, B., Keane, M.T.,1998 : Adaptation-guided retrieval: Questioning the similarity assumption in reasoning. *Artificial Intelligence* 102(2), 249-293,
- Smyth, B., McClave, P., 2001: Similarity vs. diversity. In Proceedings of the Fourth International Conference on Case-Based Reasoning. Berlin: Springer, pp. 347-361.
- Stanfill, C., Waltz, D., 1986: Towards memory-based reasoning. *Communications of the Association for Computing Machinery*, volume 29 pp. 1213-1228.
- Tartakovski, A., Schaaf, M., Maximini, R., Bergmann, 2004 R.: MINLP based retrieval of generalized cases. In Proceedings of the Seventh European Conference on Case-Based Reasoning. Berlin: Springer, pp. 404-418.
- Varma A., 1999: ICARUS: Design and Deployment of a Case-Based Reasoning System for Locomotive Diagnostics. In 3rd international conference on case-based reasoning (ICCBR-99), vol. 1650, pp. 581-595.
- Veloso.M., 1992: Learning by Analogical Reasoning in General Problem Solving. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, August.
- Watson I. et Marir F., 1994. Case-Based Reasoning: A Review. *The Knowledge Engineering Review*,