



**HAL**  
open science

## Personalized learning algorithm for diagnosis: Application to aircraft

Pierre Bect, Zineb Simeu-Abazi, Pierre-Loïc Maisonneuve, Marc Pero, Bruno  
Demerliac

► **To cite this version:**

Pierre Bect, Zineb Simeu-Abazi, Pierre-Loïc Maisonneuve, Marc Pero, Bruno Demerliac. Personalized learning algorithm for diagnosis: Application to aircraft. 24th International Conference on Condition Monitoring And Diagnostic Engineering Management (COMADEM2011), May 2011, Stavanger, Norway. pp.1352-1359. hal-00634898

**HAL Id: hal-00634898**

**<https://hal.science/hal-00634898v1>**

Submitted on 24 Oct 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## Personalized learning algorithm for diagnosis: application to aircraft

Pierre BECT<sup>1,2</sup>, Zineb SIMEU-ABAZI<sup>2</sup>, Pierre-Loïc MAISONNEUVE<sup>1</sup>,

Marc PERO<sup>1</sup> and Bruno DEMERLIAC<sup>1</sup>

1. EADS Eurocopter, Marignane- France, *first\_name.surname@eurocopter.com*

2. G-SCOP Laboratory, Grenoble - France, *first\_name.surname@g-scop.inpg.fr*

### ABSTRACT

The diagnosis of avionic equipments on aircrafts is based on recorded messages during the flight. These messages are downloaded on a ground station which is able to analyze them and provide a diagnosis to the specialized maintenance team. For a diagnosis establishment, a correlation between all messages is performed, thanks to specific rules. These rules are currently defined by an expert who uses his knowledge of the system, the avionics architecture and the feedback from customers. These rules are only based on the messages recorded during the flight. They are able to use specific characteristics of one failure code or of a grouping of failure code but they cannot take into account the environmental information. As a result, this implies the construction of more complex rules and more adapted to the system environment. This paper introduces our personalized learning process based on two learning loops. A new automatic learning algorithm uses customer feedback and takes into consideration contextual parameters relative to system's environment is proposed. This algorithm allows to design specific rules for every aircraft functioning modes.

*Keywords: adaptive algorithm, diagnosis, expert system, filtering rules, learning loop, localization rules, maintenance,*

## 1. INTRODUCTION

Nowadays, aeronautics maintenance procedure uses filtering and localization rules in order to treat avionics failure codes coming from integrated tests named Built in Test (BIT) [2]. A part of avionics failure diagnosis could be performed on board [9] and in real time [1, 10]; however this paper focuses on off-board diagnosis process.

Avionics subsystems are often designed without considering the priority issues of testability. In consequence, interactions between subsystems often generate a large number of false alarms [6]. To fill this gap, it is necessary to have a global approach for diagnosis [4]. A method is proposed in [9] which consists in combining failure codes to provide accurate rules for diagnosis.

Many researches were performed on filtering and localization rules. In [6] and [7], dynamic fault tree is applied to alarm filtering. The same process can be used to locate a failure [5]. In some studies, these rules combine temporal data relative to the occurrence of failure codes [3, 4, 6] and their characteristics [7].

The defined rules can treat the majority of cases, but, their use raises several problems:

- The first problematic is the structure of a rule. By definition, a rule is strict. When it was defined, the possible evolution was not considered. It is necessary to define a new type of rules which evolve with the use. Reference [2] includes contextual information into avionics system diagnosis. The new rule definition must take into account this type of information.
- The second critical point is the redaction of these rules. They are written by an expert who uses his knowledge of the system, particularity of avionics architecture and feedback from customers. With the time and the expansion of feedbacks, this type of treatment will become too difficult. The construction of an automatic redaction could be useful for expert and customer.

This paper presents a method which uses feedback from customers to generate and adapt rules which take into account the real use of aircraft.

After the presentation of the diagnosis principle, section 3 details a global description of the learning process. Section 4 is dedicated to the local learning loop. The deported loop is detailed in section 5 with a presentation of redaction modules. Conclusions with perspectives end this paper.

## 2. DIAGNOSIS PRINCIPLE

A diagnosis procedure is performed after each landing of aircraft. All recorded data are downloaded from aircraft in order to be analyzed. Each field of expertise has its own treatment procedure. This paper only deals with the diagnosis of avionics failures.

After the downloading process, this procedure is divided in four steps, as presented in FIGURE 1:

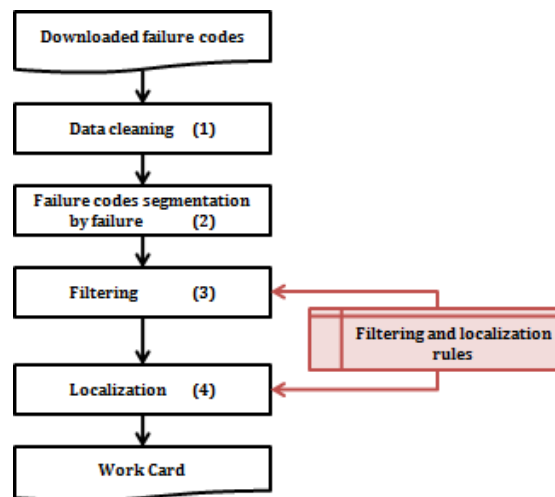


FIGURE 1. Diagnosis principle

- Step 1: Data cleaning (1):

On aircraft, every electronic equipment is redundant, and it the same for failure codes. These redundancies have to be deleted in order to facilitate the treatment. Data cleaning consists in filtering untimely failure code (redundancy filtering rules). This step is also present in local and deported learning algorithms.

- Step 2: Failure codes segmentation by failure (2):

It is an organizing step. According to the characteristics like the apparition time, the disappearing time, the duration... failure codes are grouped into "set of segment". It makes possible the application of filtering and localization rules for each segment. Like "data cleaning", this step is also used in the local and the deported learning algorithms.

- Step 3: Filtering (3):

Different filtering rules are applied in order to delete part or totality of each set of segment.

- Step 4: Localization (4):

The localization uses rules on a set of segment. This stage determines the faulty equipment(s). This step provides the appropriate work card (relative to the defective equipment) to maintenance operator. This work card helps the operator during the maintenance operation.

Rules of filtering and localization are associated to the various functions. Thus, one can distinguish two types of rules: those related to the redundancy, and those related to the structure defined in phase of design.

The following classification of different rules is proposed:

$$\begin{aligned}
R &= R_{\text{redundancy}} + R_{\text{design}} \\
R_{\text{design}} &= R_{\text{filtering}} + R_{\text{localization}} \\
R_{\text{filtering}} &= \{R_{f1}, R_{f2}, R_{f3}, \dots, R_{fi}\} \\
R_{\text{localization}} &= \{R_{l1}, R_{l2}, R_{l3}, \dots, R_{lj}\}
\end{aligned} \tag{1}$$

Where:

- $R$  = all rules
- $R_{\text{redundancy}}$  = redundancy filtering rules
- $R_{\text{design}}$  = rules which are established during designing period
- $R_{\text{filtering}}$  = filtering rules
- $R_{\text{localization}}$  = localization rules

Redundancy rules are integrated in step 1 and they do not evolve. The rules from design are built at the conception of the systems. They are composed of the filtering and localization rules which are used in step 3 and step 4. At design phase, a rule comports two parts:

$$\begin{aligned}
R_{\text{design}} &= \{C_{\text{failure\_codes}}, I_{\text{equipment}}\} \\
I_{\text{equipment}} &= \{ID_{\text{equipment}}, p\}
\end{aligned} \tag{2}$$

Where:

- $C_{\text{failure\_codes}}$  = failure codes combination which triggers the use of the rule
- $I_{\text{equipment}}$  = equipment incrimination
- $ID_{\text{equipment}}$  = equipment identification
- $p$  = weight (failure probability) which is associated to the equipment

A rule is an association of combination failure codes and equipment incrimination. The equipment incrimination is composed of the equipment identification and the weighting relative to its failure probability.

In order to improve efficiency, it is necessary to establish statistics on each rule. So, in exploitation rules integrate a counter and the weight of incriminated equipment is incremented:

$$R_{\text{exploitation}} = \{R_{\text{design}}, n\} \tag{3}$$

Where:

- $R_{\text{exploitation}}$  =  $R_{\text{design}}$  in operational context
- $n$  = number of time the rule was used

The next section presents the global learning process.

### 3. GLOBAL DESCRIPTION

The learning algorithm allows evolution and improvement of the filtering and localization rules. The operation process has to share its information with learning algorithm. This remark has an impact on the learning function. The learning process runs in parallel with the operational process. FIGURE 2 presents the two loops, one in local at customer level and the other at supplier level. This method combines advantages of the two loops: the speed at customer level and the accuracy of the supplier level.

At customer level, there is no expert control. Algorithm improves filtering and localization rules according to the customer aircraft use. It uses the same inputs as the diagnosis process and correlates them with the performed maintenance operation in order to adjust the weight which is associated to the faulty equipment. This adjustment is performed after each new maintenance procedure.

At supplier level, the *Data Collection* process collects all data from each customer. Consequently, the learning algorithm has a large amount of data. The process, described in section 5, allows experts to create new rules. These new rules are dispatched to customers by regular updates.

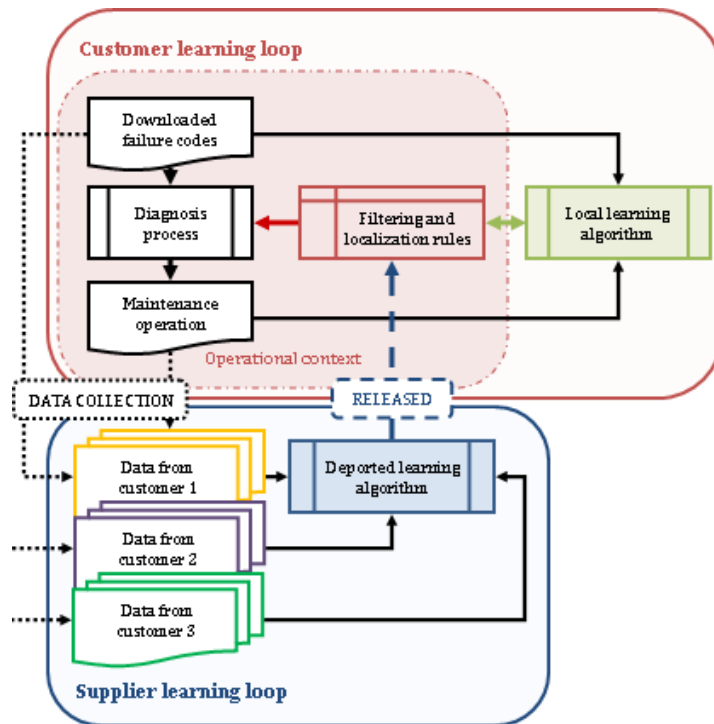


FIGURE 2. Global learning process

#### 4. LOCAL LEARNING ALGORITHM

As mentioned in the previous part, local learning algorithm runs in parallel with the operational process (FIGURE 2). The local learning loop could update the incriminations of localization rules. In order to perform this update, we correlate used rules with maintenance operations performed (FIGURE 3).

Local learning loop is composed of 4 different steps. Two of them are common with diagnosis principle: data cleaning and failure codes segmentation. Determination of matching rules (1) is the operation which consists in looking for the applicable rule in database. The failure code combination obtained with “failure codes segmentation by failure” is compared to the failure codes combination from each rule. Step (1) aims at finding the most efficient rule adapted to the failure codes combination treated.

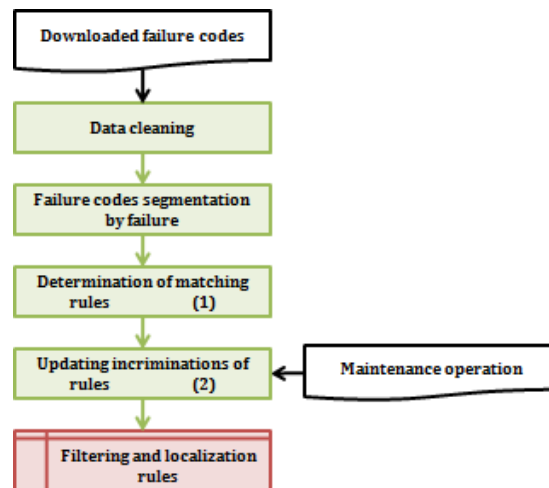


FIGURE 3. Local learning loop algorithm

Exploitation rules are composed of three different parts (equation 3): failure codes combination, equipment incrimination and counter. Local learning algorithm impacts two of these parts: counter and the weight of the incriminated equipment. Updating incriminations rules (2) correlates operation maintenance and used rule in order to increment counter and weight of the replaced equipment. This process could be illustrated with a real example (FIGURE 4):

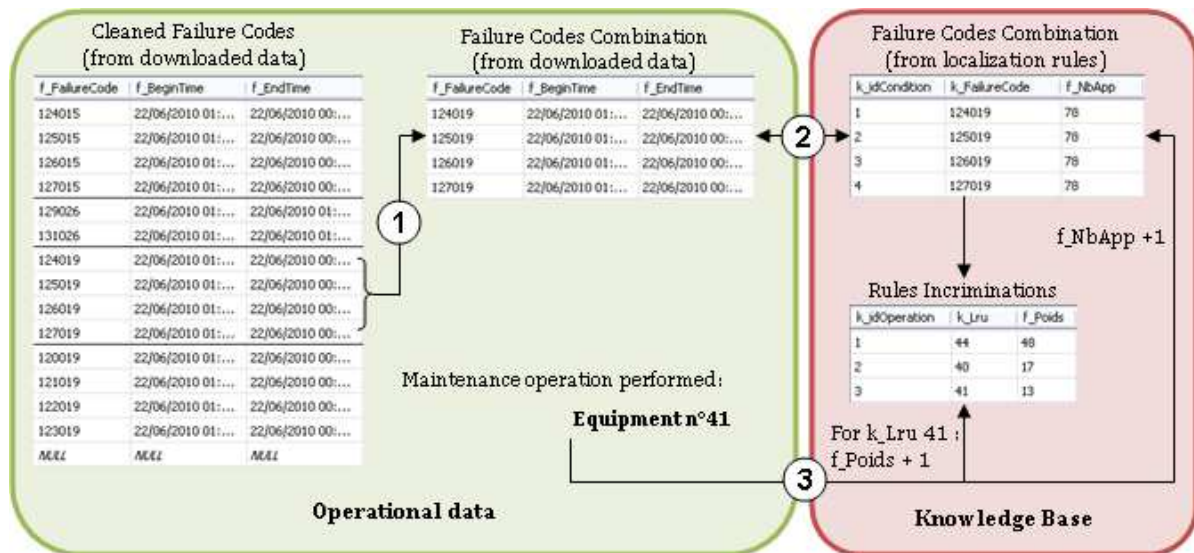


FIGURE 4. Data processing of local algorithm

The three steps of FIGURE 4 could be identified on FIGURE 3:

- **1** corresponds to the failure segmentation step. This step uses the time stamp of each failure codes in order to aggregate fault messages relative to the same failure.
- **2** identify common failure codes combination between the set of segment from downloaded data and localization rules. This step allows algorithm to find out the most adapted localization rule.
- **3** takes into account the maintenance operation performed and adjusts the weights of the rule incriminations.

This loop is an automatic process without human action. In consequence, no rules could be created in local. This task is involved in the deported learning algorithm.

## 5. DEPORTED LEARNING ALGORITHM

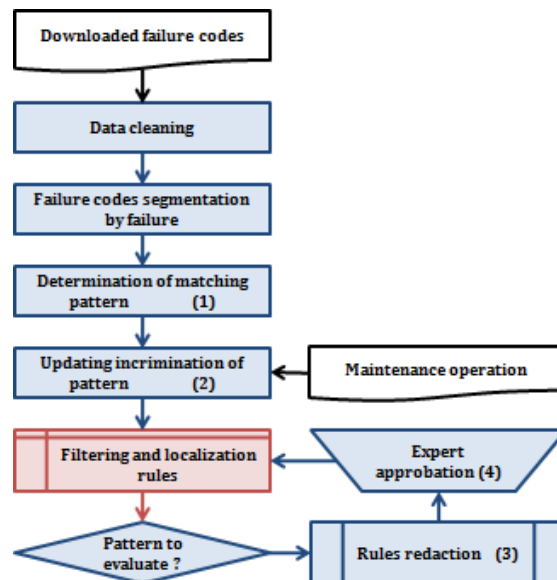


FIGURE 5. Deported learning loop algorithm

The two loops are very similar in the data treatment (FIGURE 5), even if outputs are different. Local loop provides adjustment of incriminations while deported loop designs new rules. As a consequence, deported loop treats new failure codes combination. In step (1), if no matching rules were found, set of segment is conserved and composed a part of a "pattern". The other part of this pattern is identified in step (2), when the maintenance operation sends back information. It provides the faulty equipment.

Thanks to this correlation, a new type of non applicable rule has been designed. It is called pattern. Thereafter, pattern is treated like a common rule in the learning process.

In order to assess the relevance of a pattern and to check the efficiency of the rule, a redaction module is designed.

### 5.1. Redaction module

Up to now, expert did not have tool to treat customer feedback (FIGURE 6). They create rules exactly as in the design phase. Informatics tools were designed in the past [8]. They facilitate the rules redaction but they were not able to treat the customer feedback.

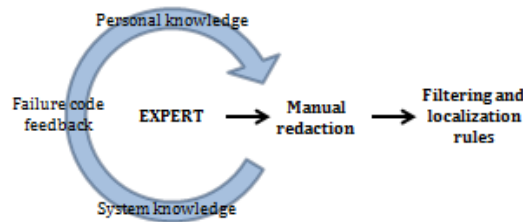


FIGURE 6. Current rules designing process

Deported learning loop is able to model pattern by using this customer feedback. These patterns need an expert assessment before their implementation. That is why; the deported learning loop includes a redaction module. This module formats the pattern to enable the validation, the modification or the revocation by the expert (FIGURE 7).

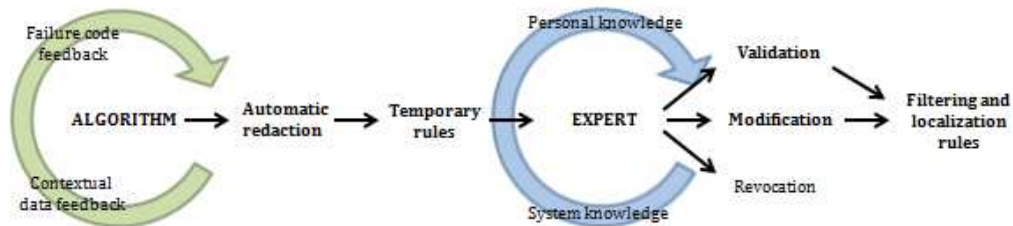


FIGURE 7. Proposed rules designing process

This redaction module uses statistics for the evaluation of the pattern relevance. A pattern is subject to assessment if:

- It has been met over  $A$  times
- Less than  $B$  different equipments have been replaced in  $C$  % of cases

$A$ ,  $B$  and  $C$  are parameterized. Setting rules were implemented in order to avoid disorder and to facilitate expert appreciation.

If a rule was considered as valid, expert must handle explicit information: the failure codes combination, the apparition order, the disappearance order and their associated maintenance operations. The redaction module provides this different information.

## 6. CONTEXTUAL DATA INTEGRATION

The customer feedback provides lot of data, coming from different climatic regions, from different customers, with different mission profiles... With more data, it becomes possible to realize more treatments. Contextual data will be complementary information for rules [2], but it is necessary to identify the needed and influent parameters.

Environmental data provide a higher level of information. It characterizes the global atmosphere of utilization with several characteristics: customer location, climatic environment, missions profiles... The characterization of the environment must be separated according to the customer studied. It becomes possible to define specificities of each "aircraft workspace".



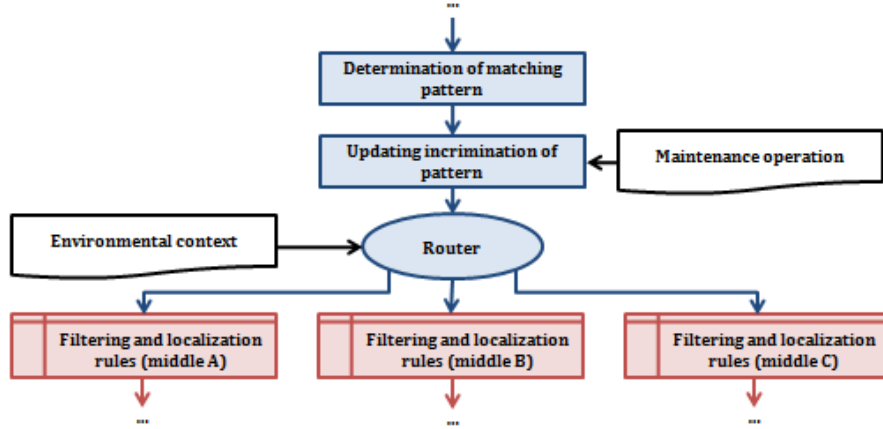


FIGURE 8. Contextual data integration

Implementation of this kind of information is not directly possible in a rule. But it could be apply on a set of rules (FIGURE 8). It is possible to define a set of rules in a specific environment. The utilization of a set of rules depends on the global utilization of aircraft (mission profile) and its operating area. FIGURE 9 presents the process and details the data flow:

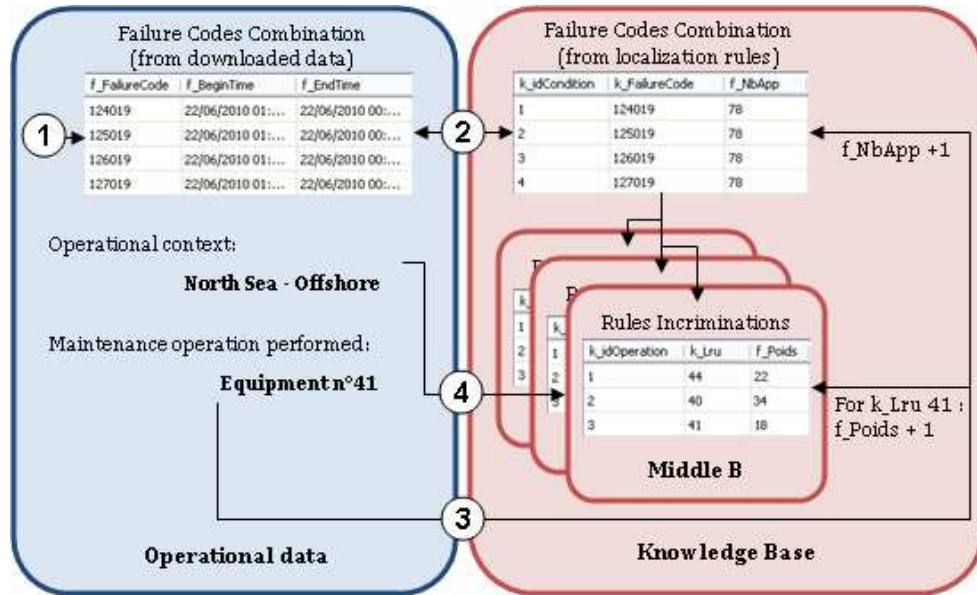


FIGURE 9. Data processing of deported algorithm

The FIGURE 9 is based on same example as the one presented in FIGURE 4. Deported learning loop provides another dimension to the learning process. 4 allows algorithm to integrate the environmental context into its localization rules.

For instance, an aircraft which operates in the North Sea has not the same constraint as an aircraft which operates in the Sub-Saharan desert. The North Sea and the Sub-Saharan desert constitute two different environments. Both aircrafts will dispose of two different sets of rules:

$$\begin{aligned}
 R_{middle\_A} &= R_{redundancy} + R_{unspecific} + R_{specific\_middle\_A} \\
 R_{middle\_B} &= R_{redundancy} + R_{unspecific} + R_{specific\_middle\_B}
 \end{aligned}
 \tag{4}$$

Where:

- $R_{middle\_X}$  = rules set attached to a specific environmental context
- $R_{unspecific}$  = rules which are not impacted by environmental data
- $R_{specific\_middle\_X}$  = specific rules attached to a defined environmental context



## 7. CONCLUSION

This paper presents rules learning algorithm based on two different loops. It is based on the improvement rules and design rules. These rules are used in avionic diagnosis process for the filtering of false alarms and for the localization of faulty equipment.

Learning loops presents several advantages:

- For customer:
  - He benefits from a local loop speed which adjusts the weight of equipment incrimination. He quickly sees the learning effects.
  - Regular releases from supplier are adapted to his own situation and to the real utilization of the aircraft. Performances of rules are improved.
- For supplier:
  - The redaction module facilitates expert job. Expert can focus on the efficiency of proposed rules and their applicability.
  - Having different rule sets allows supplier to provide different services.
  - Adapted rules and regular releases improve customer satisfaction.

Currently at Eurocopter, deported learning loop runs in a prototyping environment with the EC175 aircraft prototypes data.

## 8. REFERENCES

- [1] Bernard JP, Durocher D. 1994. An Expert System for Fault Diagnosis Integrated in Existing SCADA Svstems. Transactions on Power Systems, vol. 9 N°1: 548-514.
- [2] Fenton B, McGinnity M, Maguire L. 2002. Fault diagnosis of electronic systems using artificial intelligence. IEEE Instrumentation & Measurement Magazine 2002.
- [3] Gharib T, Nassar H, Taha M, Abraham A. 2010. An efficient algorithm for incremental mining of temporal association rules. Data and knowledge Engineering, 69: 800-815.
- [4] Kim JH, Seong PH. 2007. The effect of information types on diagnosis strategies in the information aid. Reliability Engineering and System Safety; 92: 171-86.
- [5] Simeu-Abazi Z., Di Mascolo M., M. Knotek " Fault diagnosis for discrete event systems: modelling and verification ", Reliability Engineering and system safety, Volume 95, Issue 4, April 2010, Pages 369-378
- [6] Lefebvre A, Simeu-Abazi Z, Derain JP. 2010. A methodology of alarm filtering using dynamic fault tree, Reliability Engineering & System Safety - Volume 96, Issue 2, February 2011, Pages 257-266
- [7] Lefebvre A. 2009. Contribution à l'amélioration de la testabilité et du diagnostic de systèmes complexes : application aux systèmes avioniques. Thesis of Joseph Fourier University, Grenoble.
- [8] Minakawa T, Ichikawa Y, Kunugi M, Shimada K, Wada N, Utsunomiya M. 1995. Development and implementation of a power system fault diagnosis expert system. Transactions on Power Systems, vol 10 n°2: 932-940.
- [9] Schneider N, Delprat S. 2010. Method for recognising sequential patterns for a method for fault message processing. International patent from EADS.
- [10] Vasquez E, Chacon OL, Altuve H. 1997. An on-line expert system for fault section diagnosis in power Systems. Transactions on Power Systems, vol 12 n°1: 357-362.