A Decision System for Aircraft Faults Diagnosis based on Classification Trees and PCA

ZeFeng WANG
Institut des Systèmes Intelligents et de Robotique
Université Pierre et Marie Curie (Paris VI)
4 Place Jussieu, Paris 75005, France
zwang@isir.upmc.fr

Jean-Luc ZARADER, Sylvain ARGENTIERI, Karim YOUSSEF
Institut des Systèmes Intelligents et de Robotique
Université Pierre et Marie Curie (Paris VI)
4 Place Jussieu, Paris 75005, France
zarader@isir.upmc.fr; argentieri@isir.upmc.fr; youssef@isir.upmc.fr

***Abstract* — Aircrafts are complex systems that require permanent and precise monitoring and troubleshooting. The automation of these tasks is thus of a high importance. This paper presents an intelligent decision system for faults diagnosis of aircrafts. The system relies on decision trees, being easier to interpret, quicker to learn than other data-driven methods, and able to work even with missing pieces of information. The used C4.5 algorithm automatically "learns" the best decision tree by performing a search through the set of possible trees according to the available training data. And Principal Component Analysis (PCA) is used to decrease the input data's dimension. Compared to other methods, the proposed one is more advantageous and some presented evaluations demonstrate its abilities. High correct faults detection rates and low missed detection and false alarm rates are obtained. Such a decision system is highly useful for engineering consulting services, accumulating the knowledge for the operational rules of diagnosis, and the design of new aircrafts.**

Keywords: Faults Diagnosis, Aircrafts, Decision System, Decision Tree, PCA.

I. INTRODUCTION

Faults diagnosis is of a high importance for modern aircrafts, with structures and systems becoming more and more complex. Hundreds of sensors are being used to supervise an aircraft, but abnormal information and faults are still difficult to find. In order to deal with this problem, many faults diagnosis systems have been invented, with intensive studies of data mining. One can cite: the expert system [1] that needs to establish the knowledge base and the rules by experts and has to face an intractable problem - the conflict of rules [2]. Also, neural networks [3] and Support Vector Machines (SVMs) [4] which require much more time for the learning process.

In this paper, a novel intelligent decision system for faults diagnosis of aircrafts is proposed. We decided to use a decision tree learning algorithm since it provides much easier data to interpret than other algorithms, such as neural networks and SVMs [5]. Decision tree learning is a method for approximating discrete-valued target functions [6]. A decision tree algorithm automatically "learns" a decision tree by performing a search through the space of possible trees to find the one that best fits the training data. The particular algorithm used in this paper is known as C4.5 (see Section II). As Figure 1 shows, the decision system uses the output information of sensors, and its diagnosis results are delivered to the fault tolerance system and the decision system which will demonstrate the faults and suggest decisions to the pilot and the ACC (airport command center) (emergency landing, return to base or mission continuation if the fault tolerance system can deal with the fault, etc.). In accordance with the results of the decision system, the pilot or ACC can decide their next action.

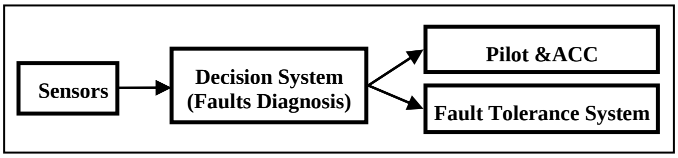

Figure 1. **Overall Diagnosis System**

Aside the normal working conditions of the aircraft and thus of the fault diagnosis system, the real risk events in the system should be taken into consideration during a flight. Such cases can be seen for example in the loss of some sensors' information in special situations, especially for the fighting aircrafts. And in these cases, the diagnosis system should be ensured of working continuously. The presented system is able to deal with these cases. This paper studies the case where 3 faults of the anti-icing system of the aircraft X (Dassault Aviation) need to be diagnosed. Usually the hot air exhausted by the aircraft's engine is used in the anti-icing system to de-ice the aircraft wings. 33 sensors supervise the anti-icing system. They mainly provide values of the engine and air wings' temperatures, pressures and so on. So our diagnosis system has input vectors of a dimension 33 used to analyze and build the decision tree. Note that the natures of the faults cannot be revealed in this paper because of confidentiality issues.

The rest of this paper is organized as follows. Section II demonstrates the classical decision tree fundamentals, and shows a study of the tree's over fitting and the pruning of unnecessary branches. Section III demonstrates the usage of PCA in decreasing the initially high input data dimension. Both the procedure of that application and its effect on the system are included in this section. Section IV reveals the robustness and the fault-tolerance of the system even in the case where some sensors are lost. Section V shows the results of

DGCIS, Conseil régional d'Ile de France, Conseils généraux 77 et 78 (sponsors).

evaluations studying the cases presented before; and the last section concludes the paper and presents a preview of future works.

II. DECISION TREE

A. Decision Tree Fundamentals

Decision tree learning is a method commonly used in data mining. Its goal is to create a model that predicts a target value (that can be the class of an input example), based on several input values that describe the example. This method uses a hierarchical tree structure that progressively checks input variables and makes corresponding decisions like classification. Like the system of expert, learned trees can also be re-represented as sets of if-then rules to improve human readability, but they don't need human experts to establish a knowledge and rules database, they build rules by themselves.

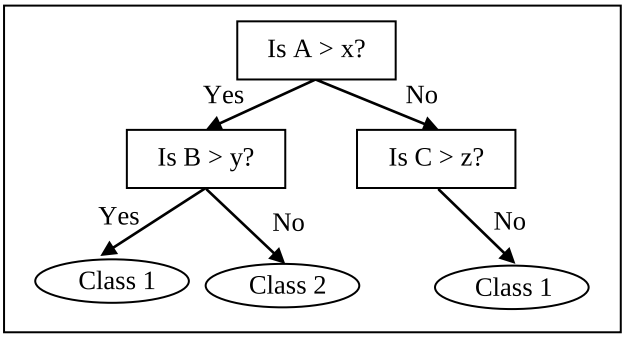

Figure 2**. A decision tree used to discriminate 2 classes with inputs consisting of 3 values (A, B and C).**

An example is shown in Figure 2. Data represented by three input variables can be classified into two classes. Each node corresponds to one of the input variables – A or B or C, and inside each node, the variable is compared to a corresponding edge. The result of this comparison is translated by a leaf that leads to another node, or to the final classification decision. But in the presented example, one can ask several questions, like: why is it that A is in the root, not B or C? How was the threshold x found for A? So, in a decision tree algorithm, and in a more general manner, two questions are to be answered, in order to find the best possible classifier: 1) How to decide the sequencing of variables and corresponding nodes in the tree? 2) How to determine the used thresholds values?

In order to answer the first question, and thus to determine the best sequencing of the variables, we resort to a measurement that is commonly used in information theory, called entropy. Entropy characterizes the impurity of an arbitrary collection of examples [7]. For example, given a collection S containing positive and negative examples of a target concept, the entropy of S relative to this Boolean classification is:

$$Entropy(S) = -P_{\oplus}\log_2 P_{\oplus} - P_{\Theta}\log_2 P_{\Theta} \qquad \textbf{(1)}$$

where $P_{\oplus}$ and P_{Θ} are respectively the proportions of positive and negative examples in S. Note that in calculations involving entropy we consider 0* log (0) to be equal to 0. In our case, a collection will be the set of values provided by a sensor, that is to say, a variable. Such a variable can be Boolean, like in the example, or can take multiple values. So at first, we calculate the entropies of all the variables. Then, a measurement of the effectiveness of an attribute or value that a variable can take is needed. We use a measure called information gain that is simply the expected reduction in entropy caused by partitioning the according to the attributes and values that a variable can take. More precisely, the information gain, Gain(S, A), of an attribute A, relative to a collection of examples S, is defined as:

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \qquad \textbf{(2)}$$

where values(A) is the set of all possible values for the variable A, and S_v is the subset of S for which the variable A has the value v. So, as said before, at first the entropies and the information gains of all the variables are calculated. The variable with the highest information gain is the root node from which leaves go out, each leaf corresponds to a value that its variable can take. Following this, entropies and information gains corresponding to the other variables and to each of the leaves going out of the root nodes are computed, and new nodes are taken corresponding to the highest gains. And so on, until all variables are used and the final tree is formed. Classification is done by using the values of the variables constituting an example and following the resulting tree leaves and nodes from the root to the final leaf which gives the class.

In this system, all the input variables are not Boolean but can take multiple values. Therefore, the values of each attribute are sequently arrayed firstly, and then all the expected pair values which are neighboring but sorted in different classes are picked out. After that, all possible threshold values are calculated, as the halves of the sums of the pairs mentioned before. Finally, the threshold value with maximum information gain is taken.

B. Tree pruning to avoid over fitting

Cross Validation (CV) is used to determine how deep the decision tree will grow. This is known as the post-pruning rule. Being used by the C4.5 algorithm [8], it involves the following steps:

1. Use the CV to infer the decision tree from the training set, grow the tree until the training data is as fit as possible and allow over fitting to occur.

2. Convert the learned tree into an equivalent set of rules by creating one rule for each path from the root node to a leaf node.

3. Prune (generalize) each rule by removing any preconditions that result in improving its estimated accuracy.

4. After sorting the pruned rules based on their estimated accuracy, involve them into classification of subsequent instances.

So in summary, CV performs a search through multiple pruned tree configurations, and keeps the configuration that gives the best validation results. In this paper, the decision system is built with 33 sensors' information. As shown in

Figure 3, CV operated during a learning procedure allowed us to find the best size of the tree based on these 33 sensors, with 11 terminal nodes. This is obtained while the number of nodes without CV can actually grow much higher than 11. Figure 4 shows the pruned tree, originally 59 nodes are obtained without pruning, the dotted lines are the leaves to prune, and the remaining nodes contain the rules that the system finally used. It is clearly that there is one root with 14 branches in the original, and only 6 branches left after optimization, which highly enhances the working efficiency.

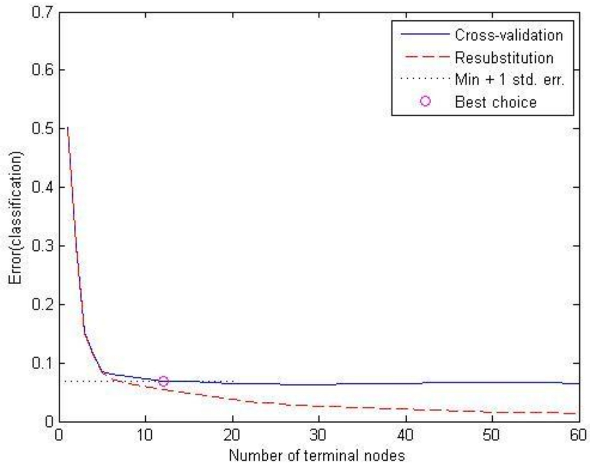

Figure 3. Using Cross-Validation to find the optimal tree size

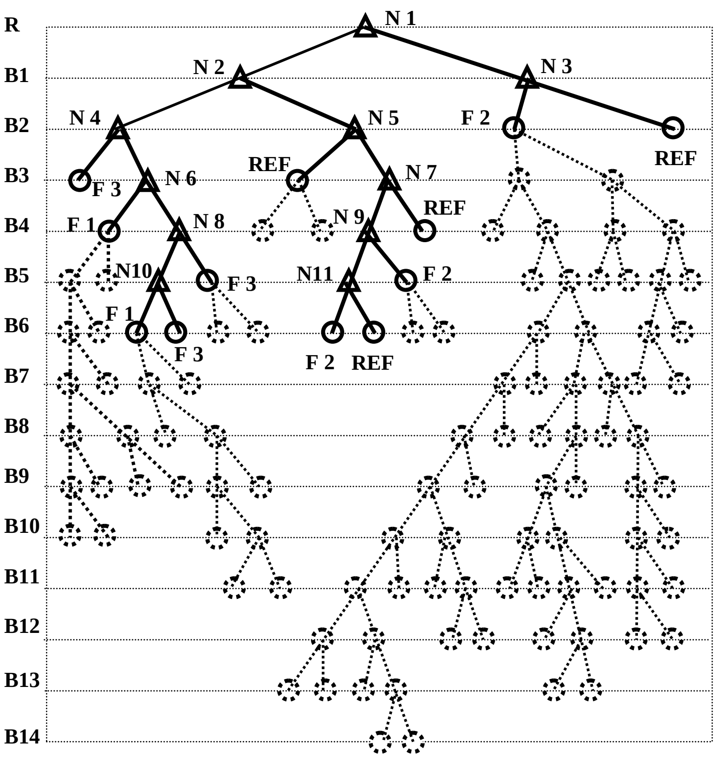

Figure 4. Diagnosis tree and leaves to prune
R: Root; B: Branch (B1-B14); N: Node (N1 – N11);
F1: Fault 1; F2: Fault 2; F3: Fault 3; REF: No-Fault.

III. PRINCIPAL COMPONENT ANALYSIS

For the sake of safety, plenty of the sensors installed in the aircraft are redundant. Multiple sensors can do the same role, in order not to lose the information when one of them malfunctions. Indeed, the sensors' redundancy is indispensable for the aircraft, especially with sensitive and important parts. But for the diagnosis system used here, using all the sensors' data directly in the input space at once is not very necessary and it affects the arithmetic speed of the diagnosis system. Principal Component Analysis (PCA) is employed, as being a tool that effectively reduces the input data dimension and improves the computational loads of the diagnosis process, and thus of the overall system. PCA is mathematically defined [9] as an orthogonal linear transformation that converts a set of observations of possibly correlated variables into a set of values of uncorrelated variables called principal components. The data is projected to a new coordinate system where the greatest variance of the data obtained by any projection comes to lie on the first coordinate (principal component). The second greatest variance lies on the second coordinate, and so on. The number of principal components is generally less than the number of original variables. So PCA reduces the dimension of our aircraft diagnosis system input data. For example, in our case, Figure 5 shows that the first principal component's variance constitutes more than 50% of the total sum of all variances over all components. This variance decreases as the order of the corresponding component increases. Finally, 6 principal components were found to add up to 95% of the total sum of the variances in our case. Note that these principal components are computed based on the 33 used anti-icing system sensors and thus a big dimension reduction is performed, while keeping most of the beneficial information.

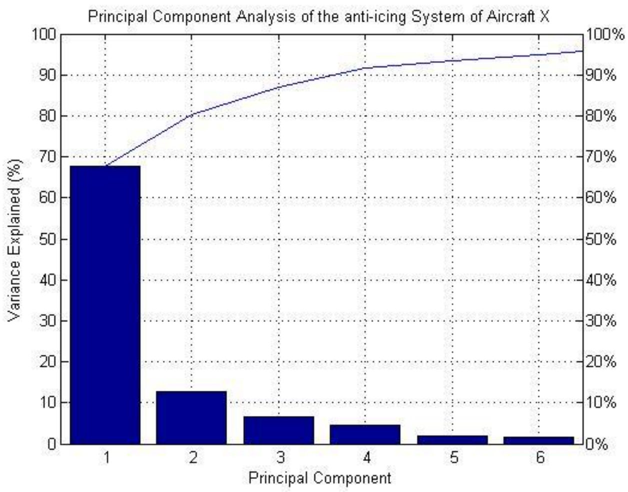

Figure 5. PCA of the anti-icing system of the aircraft X: 6 first principal components, their corresponding variances and the cumulative variance.

IV. INFORMATION LOSS AND REBUILDING OF THE TREE

In the real conditions of an aircraft's flight, especially with the fighting aircrafts, there is a risk of losing sensors or sensors' information. This loss of information is harmful for the diagnosis system's operation as it can lead to inadequate responses. The problems of such cases especially arise with diagnosis systems using neural networks or SVMs, where the absence of signals or the zero-valued inputs lead to inactivity of certain parts of the networks. However, while the diagnosis tree cannot resolve this problem, it can perform a quick relearning based on the initial learning database and using only the data of the currently active sensors. For example, with a computer – Intel Pentium Dual-Core T4400 and 2G memory, it only needs 46 seconds for relearning with a 33x160000 dataset while with an SVM or a neural network it takes more than one hour.

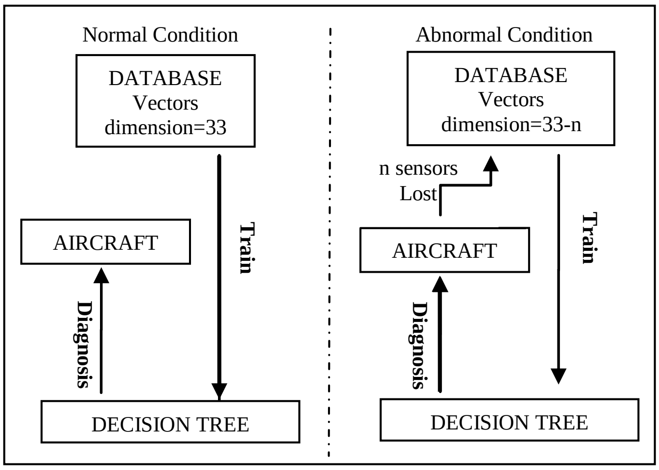

Figure 6. Relearning of the diagnosis tree

Figure 6 shows data provided by 33 sensors and m learning examples. If the aircraft loses n sensors or sensors' information, the diagnosis tree will automatically detect the loss and get relearned using new vectors of dimension 33-n. This ensures an appropriate online functioning of the system during a flight.

V. RESULTS

A. Experimental procedure

A database consisting of 160000 examples (160000 records, one record per second) is used to build the diagnosis system and test it. It is constituted of data vectors obtained with the pre-described 33 sensors of the anti-icing system of the aircraft X. Data is measured during a flight and describes the faults and the normal aircraft state without faults. 75% of the data were chosen randomly for the training and the remaining 25% are used to test the tree.

During the training, CV is used to avoid the over fitting problem. Subsection B shows the PCA's utility in reducing the data dimension, subsection C simulates a dangerous situation of sensors loss and presents corresponding results. In order to get a valid scientific and compelling evaluation scheme, a trail of one hundred repetitions of all the processes is studied as shown in Figure 7. The evaluations study the ratios of the following criteria: Correct Decision, Missed Detection, and False Alarms as shown in the last two subsections.

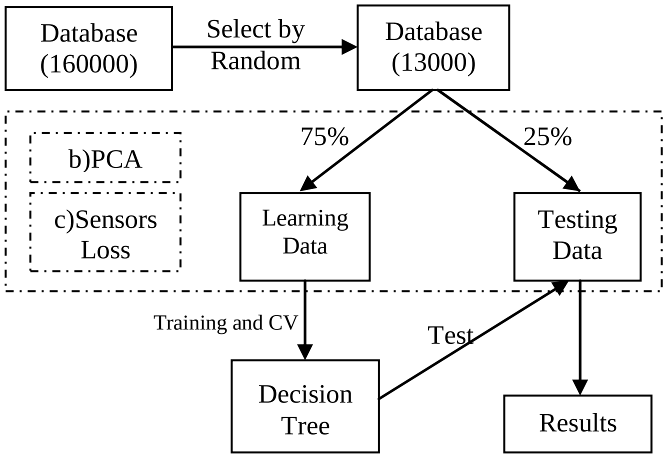

Figure 7. Experiments Drawing

B. Results in the normal Condition

The test results will be presented in two forms corresponding to two different conditions: the condition of maintenance and the condition of mission on-line.

1) In the condition of maintenance, the system must distinguish clearly each fault and show the diagnosis results to an engineer. With the Confusion Matrices which are shown in the Tables I - IV, three observations can be reached:

a) *Performance of system with PCA:* Although PCA reduces the dimension of the data, the performance of the decision system is not reduced. The score of the system based on PCA is nearly the same as that of the system without PCA, especially for the systems with a pruned tree. For example, the score of Table II is 95.16% and the score of Table IV is 95.17%.

b) *Performance of system based on Pruned Tree* : In comparison with Table I and Table II (or Table III and Table IV), as mentioned in Section II, the results confirm that a system with a pruned tree works better than a system with a tree without pruning. The pruned trees add about two to four percentage points to the scores of diagnosis.

c) *Research of faults with confusion matrix:* Some words about data mining of Aircraft X's anti-icing system: as shown in these tables, the fault 2 and fault 3 are close to each other and Ref and Fault 1 are similar. In these tables, the system never confused Ref or Fault 1 with Fault 2 or Fault 3, and vice versa. In addition, Fault 2 can be perfectly diagnosed by a system with a pruned tree.

TABLE I. CONFUSION MATRIX, TESTING WITH A WHOLE TREE (WITHOUT PRUNING), SCORE = 93.64%

Target/Output	**REF**	**Fault 1**	**Fault 2**	**Fault 3**
REF	93.67%	6.33%	0.00%	0.00%
Fault 1	2.61%	97.39%	0.00%	0.00%
Fault 2	0.00%	0.00%	94.59%	5.41%
Fault 3	0.00%	0.00%	11.11%	88.89%

TABLE II. CONFUSION MATRIX, TESTING WITH A PRUNED TREE, SCORE = 95.16%

Target/Output	REF	Fault 1	Fault 2	Fault 3
REF	98.73%	1.27%	0.00%	0.00%
Fault 1	6.96%	93.04%	0.00%	0.00%
Fault 2	0.00%	0.00%	100.00%	0.00%
Fault 3	0.00%	0.00%	11.11%	88.89%

TABLE III. CONFUSION MATRIX, TESTING WITH A WHOLE TREE (WITHOUT PRUNING), AND PCA = 95% (WITH 6 VECTORS), SCORE = 91.65%

Target/Output	REF	Fault 1	Fault 2	Fault 3
REF	94.50%	5.50%	0.00%	0.00%
Fault 1	8.14%	91.86%	0.00%	0.00%
Fault 2	0.00%	0.00%	92.11%	7.89%
Fault 3	0.00%	0.00%	11.76%	88.24%

TABLE IV. CONFUSION MATRIX, TESTING WITH A PRUNED TREE, AND PCA = 95% (WITH 6 VECTORS). SCORE = 95.17%

Target/Output	REF	Fault 1	Fault 2	Fault 3
REF	93.58%	6.42%	0.00%	0.00%
Fault 1	1.16%	98.84%	0.00%	0.00%
Fault 2	0.00%	0.00%	100.00%	0.00%
Fault 3	0.00%	0.00%	11.76%	88.24%

2) In the condition of mission on-line, the pilot doesn't need to know exactly which type of fault of anti-icing system appeared, he only needs to know that the anti-icing system is broken or not. If it doesn't work, he must go back to ACC or make a emergency landing. Therefore, all the types of faults are grouped as one fault in this condition, the correct detection ratio, missed detection ratio and false alarm ratio will be studied in this part.

TABLE V. TESTING RESULTS WITH A TREE WITHOUT PRUNING

CASE	Amount of vectors	Correct Detection	Missed Detection	False Alarm
Without PCA	33	96.49%	3.51%	4.42%
PCA 99%	12	96.04%	3.96%	5.54%
PCA 95%	6	95.49%	4.51%	5.52%
PCA 90%	4	90.73%	9.27%	7.84%
PCA 80%	2	85.06%	14.94%	8.96%

TABLE VI. TESTING RESULTS WITH A PRUNED TREE

CASE	Amount of vectors	Correct Detection	Missed Detection	False Alarm
Without PCA	33	97.43%	2.57%	4.16%
PCA 99%	12	97.08%	2.92%	4.76%
PCA 95%	6	95.62%	4.38%	4.76%
PCA 90%	4	92.95%	7.55%	5.45%
PCA 80%	2	88.83%	11.17%	5.96%

As can be seen in Table V and Table VI, the system offers satisfying performances with high correct detection rates, and low missed detection and false alarm rates. The performances slightly decrease with decreasing input data dimensions, but the results remain highly accurate. This may be due to one of two hypotheses:

a) The faults detection and discrimination remains easy for the system, even with a change of input conditions and tree structure.

b) PCA finds the principal components and the system can get well adapted to them and classify the aircraft's situations.

But, there still exists a big problem: all the data is recorded per second during the mission of the aircraft, if there is 5% missed detection ratio or false alarm ratio, the pilot will receive the missed detection or false alarm 3 times per minute, which is unacceptable in the real applications. To solve this problem, statistics of the results of the diagnosis system during a period of time, as 15 or 30 seconds, are studied. That is to say, the system records the diagnosis results from a time; then compares the fault's ratio diagnosed in the results with pre-set threshold; at last the system will judge it as fault occurred if its value is above the threshold. For example, the pre-set threshold is 15%, and only two results are diagnosed as fault by the system during 15 seconds. Since the frequency is below the threshold, the diagnosis of these 15 seconds is considered as "no-fault". This criterion is used to test the continuous data of 18000 seconds: the Correct Detection ratio equals 100%, the Missed Detection ratio equals 0, and the False Alarm ratio equals 0.0013% (using 6 vectors of input data whose PCA is about 95%).

C. *Results in the abnormal Condition*

In this subsection, a state of emergency is described: a number of sensors are lost. To simulate this, at first the number of the sensors is randomly reduced, without PCA. And then PCA is used to pre-treat the data if a few sensors like 1 ~ 5 are lost. In this case, PCA wasn't able to reduce the dimension of the input data if there were already many sensors lost. We run the system one hundred times, and at each time sensors are lost by random; at last we calculate the mean of the correct detection, the missed detection and the false alarm of the tests series.

TABLE VII. RESULTS WITH A TREE WITHOUT PRUNING IN AN ABNORMAL CONDITION

Lost Sensors	Correct Detection	Missed Detection	False Alarm
3	95.46%	4.54%	4.51%
5	93.06%	6.94%	8.10%
10	90.71%	9.29%	13.56%
15	83.84%	16.16%	18.25%

TABLE VIII. RESULTS WITH A PRUNED TREE IN AN ABNORMAL CONDITION

Lost Sensors	Correct Detection	Missed Detection	False Alarm
3	97.25%	2.75%	2.77%
5	95.85%	4.15%	5.44%
10	93.57%	6.43%	8.67%
15	87.94%	12.06%	11.03%

Table VII and Table VIII show the stability of the system within several lost captors. While the missed detection and false alarm ratios rise with the loss of sensors information, the performances of the system remain sufficiently accurate.

VI. CONCLUSION AND FUTURE WORKS

In this paper, a faults diagnosis system of aircrafts using decision trees is presented. The learning speed of this intelligent system is much higher than the speed of other diagnosis systems, like neural networks or SVMs, which allows it to perform a quick relearning in the cases of data loss. PCA is used to accurately reduce the system's input dimension. Tests results of normal and abnormal flight conditions demonstrate the high abilities of the system. For the future work, we will focus deeply on how to prognosis the situations (failure prediction) of an aircraft with a decision system.

ACKNOWLEDGMENT

All the data are supplied by Project MODIPRO "MOdelisation, DIagnosis and PROnostic".

Financiers : DGCIS, Conseil régional d'Ile de France, Conseils généraux 77 et 78.

Partners: DASSAULT AVIATION, ISIR/UPMC, IT4 CONTROL, INRIA, SupElec, SNECMA, BAYESIA, KBS.

References

[1] Chen WenBin, Liu XiaoLing, He ChangJiu, Liu YiJun, "Knowledge Base Design for Fault Diagnosis Expert System Based on Production Rule", 2009 Asia-Pacific Conference on Information Processing 978-0-7695-3699-6/09 © 2009 IEEE

[2] Chen WenBin, Liu XiaoLing, Fang Yu, Zhang Jian, "Inference Engine Design of Expert System Based on Blackboard Model and Fault Tree",.2009 Asia-Pacific Conference on Information Processing 978-0-7695-3699-6/09 © 2009 IEEE.

[3] Liao Wei, Wang Hua, Han Pu, "Neural Network Modeling of Aircraft Power Plant and Fault Diagnosis Method Using Time Frequency Analysis". 2009 Control and Decision Conference 978-1-4244-2723-9/09 @ 2009 IEEE

[4] LI Bin, ZHANG Wei-guo, NING Dong-fang, and YIN Wei, "Fault Prediction System Based on Neural Network Model", Innovative Computing, Information and Control Conference 0-7695-2882-1/07 ©2007 IEEE

[5] Mark Schwabacher, Robert Aguilar, Fernando Figueroa, "Using Decision Trees to Detect and Isolate Simulated Leaks in the J-2X Rocket Engine", IEEEAC paper #1408, Version 1, Updated November 1, 2008

[6] Arash Jamehbozorg and S. Mohammad Shahrtash, "A Decision-Tree-Based Method for Fault Classification in Single-Circuit", IEEE TRANSACTIONS ON POWER DELIVERY, VOL. 25, NO. 4, OCTOBER 2010

[7] Tom M.Mitchell, "Machine Learning", McGraw-Hill Science/Engineering/Math; (March 1, 1997)

[8] Quinlan, J. Ross. 1993. "C4.5: programs for machine learning". Morgan Kaufmann Publishers Inc

[9] Jolliffe I.T. "Principal Component Analysis", Series: Springer Series in Statistics, 2nd ed., Springer, NY, 2002, XXIX, 487 p. 28 illus. ISBN 978-0-387-95442-4