An Interpretation of Neural Networks as Inference Engines with application to Transformer Failure Diagnosis

Adriana R. Garcez Castro Vladimiro Miranda

Abstract – **An artificial neural network concept has been developed for transformer fault diagnosis using dissolved gas-in-oil analysis (DGA). A new methodology for mapping the neural network into a rule-based inference system is described. This mapping makes explicit the knowledge implicitly captured by the neural network during the learning stage, by transforming it into a Fuzzy Inference System. Some studies are reported, illustrating the good results obtained.**

I. INTRODUCTION

The correct operation of power transformers is decisive to the secure functioning of a power system. Therefore, it is of great importance to detect and eliminate transformer incipient faults before it deteriorates to a severe condition

It is known that transformer faults, mainly in the form of overheating, arcing or partial discharge, develop certain gaseous hydrocarbons, which are retained by the insulating oil as dissolved gases. The concentration, relative proportion and generation rate of these gases have been extensively used for the estimation of the condition of a transformer. Methods such as Dornenburg Ratios, Rogers Ratios and IEC Ratios are commonly used by utilities. However, the analysis of the gases and interpretation of their significance have been to some extent an art subject to variability. Therefore, the search for a more reliable method using the information on concentration of dissolved gases is still a hot topic.

Some studies have reported the efficiency and difficulties of using Artificial Neural Networks (ANNs) and Fuzzy Logic [1-3] in transformer diagnosis. In Fuzzy Systems, the proportion of gases has been fuzzified to represent the vague nature of DGA. These fuzzy systems were in general built according to DGA methods and the efficiency of the system depended on the completeness of the knowledge of the specialist. Moreover, the rules could not be automatically adjusted through a self-learning process when new knowledge is acquired.

To overcome the fuzzy systems drawbacks, artificial neural networks have been proposed to deal with transformer fault diagnosis. Although ANNs have been recognized by their powerful capacity to express relationships between the variables of a problem, there is still much distrust in them for a number of reasons. One often heard argument is that ANNs do not have explaining capability. In a number of ways, this is œrtainly true. In many cases, ANNs are sufficient and there is no real need to make knowledge explicit. But in some application areas this will be felt as a must. A good example would be in transformer fault diagnosis. Human understanding would be greatly enhanced if the relations between the variables were explicit, and engineers or technicians would also gain more confidence in the diagnoses produced.

Considering the abilities of neural networks to deal with classification problems, in this paper we propose a neural network based system for transformer fault diagnosis using dissolved gas -in-oil analysis. And to overcome the problem of the lack of explaining capabilities of a neural network, the knowledge hidden in its structure will be uncovered using a new methodology that allows mapping an ANN into a rule-base system. This transformation will make explicit the knowledge implicitly captured by the neural network trained for transformer fault diagnostic and it will allow the human specialist to understand how the neural network arrives a particular result. Neural networks no longer will be seen as a "black-boxes". Moreover, the path to the discover of new knowledge becomes open.

The results presented correspond to an evolution of a prior attempt [4], which still did not exhibit the desired property of transparency of the rule base now achieved.

II. NEURAL NETWORKS AND FUZZY SYSTEMS

An ANN is characterized by having in its architecture many low-level processing units with a high degree of interconnectivity via weighted connections. Among many ANN concepts, the most commonly used is the Multilayer Feedforward Neural Network (Figure 1).

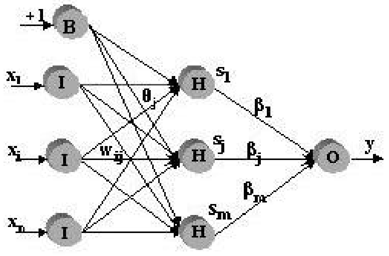

Fig.1. ANN topology

For this ANN, each neuron in a hidden layer H calculates:

$$s_j = f(\sum_{i=1}^{n} x_i w_{ij} + q_j) \qquad (1)$$

where x_i is the *i-th* input to the net, w_{ij} is the weight of the connection from input neuron *i* to hidden neuron *j*, q_j is the bias of the *j-th* hidden neuron and *f*(.) is the activation function of the neuron.

Adriana R. Garcez Castro is with INESC Porto, Portugal, and also with NESC/UFPA (Federal University of Pará, Brasil (email: acastro@inescporto.pt)

Vladimiro Miranda is with INESC Porto, Portugal, and also with FEUP, Faculty of Engineering of the University of Porto, Portugal (email: vmiranda@inescporto.pt)

For the output layer O, each neuron calculates:

$$y_k = g(\sum_{j=1}^{m} b_{jk} s_j + q_k) \qquad (2)$$

where β_{jk} is the weight of the connection from hidden neuron *j* to output neuron *k* , y_k is the *k-th* output of the net q_k is the bias of the k-*th* output neuron and *g(.)* is the activation function of the neuron.

ANNs have been considered black boxes because it has been argued that nothing can be revealed about the knowledge encoded within them. On the other hand, Fuzzy Inference Systems (FIS) or Fuzzy Rule Based Systems (FRBS), unlike ANN, are systems that have precisely the desired characteristics of an explicit form of knowledge. FIS are dynamic, parallel processing systems that estimate input-output functions.

In Takagi-Sugeno (TS) fuzzy inference systems, the relationship between variables of the system is represented by means of fuzzy IF-THEN rules in the form:

Rule R_l: IF x^l_1 is C^l_1 and … and x^l_n is C^l_n

THEN $y^l = f(x_1,..., x_n)$ (3)

where C^l_i are fuzzy sets, x_i is the input of the system.

The consequent of the rule is an affine linear or non-linear function of the input variables and the output of the TS model is computed as the weighted average of y^l.

When y^l is a constant, the fuzzy inference system is called a Zero-order TS fuzzy model

III. MAPPING NEURAL NETWORKS INTO A TAKAGI-SUGENO FUZZY MODEL

A. *Definition of the topology of the ANN*

For the purpose of this paper consider the ANN in Figure 1. This ANN has one neuron in output layer with a *linear function* as activation function and has only one hidden layer whose activation function for each neuron is the *sigmoid function*, as shown in Figure 2. This sigmoid function is defined as follows:

$$g(x) = \begin{cases} 1 - e^{-x} & x \geq 0 \\ \mathrm{e}^{x}\ \text{-1} & x < 0 \end{cases} \qquad (4)$$

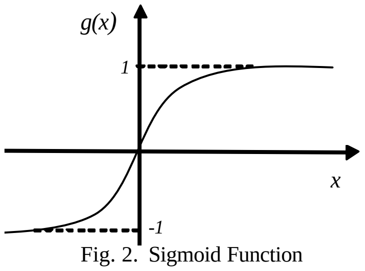

Fig. 2. Sigmoid Function

B. *Introducing the concept of f-duality*

The concept of *f*-duality was introduced by Benitez, Castro and Requena in [5]. To produce the mapping of an ANN into rule sets as proposed in this paper, this concept will be used to find the equivalent mathematical operation to equation (1) – the operation calculated for the hidden neuron.

The following proposition is useful:

Proposition 1 : Let *f*: X→Y be a bijective function and let ⊕ be an operation defined in the domain of *f*, X. Then there is one and only one operation ⊗, defined in the range of *f*, Y, verifying:

$$f(\bigoplus_{i=1}^{n} x_i) = \bigotimes_{i=1}^{n} f(x_i) \qquad (5)$$

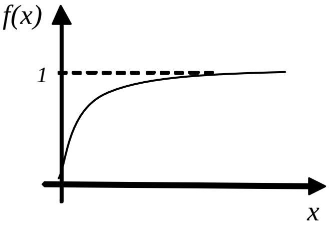

Fig.3..Positive-Sigmoid function

To apply the concept of *f*-duality let consider the sigmoid function *g(x)* defined in (4) only to $x \geq 0$. This leads to the function *f(x)* that will be called positive-sigmoid function and whose graphic is shown in Figure 3 . The positive sigmoid function is defined as follows:

$$f(x) = \begin{cases} 1 - e^{-x} & x \geq 0 \\ 0 & x < 0 \end{cases} \qquad (6)$$

Therefore, applying the concept of *f*-duality to (1), without bias q_j , where the activation function *f* is now as defined in (6) and considering $\sum_{i=1}^{n} x_i w_{ij} \geq 0$, then the output signal of the hidden neurons can also be calculated by:

$$s_j = f(\sum_{i=1}^{n} x_i w_{ij}) = f(x_1 w_{1j}) * ... * f(x_n w_{nj}) =$$
$$1 - (1 - f(x_1 w_{1j}))...(1 - f(x_n w_{nj}))$$
$$\text{if } \sum_{i=1}^{n} x_i w_{ij} \geq 0 \text{ and } x_i w_{ij} \geq 0 \qquad (7)$$

Noticing that the function *f(* $x_i w_{ij}$ *)* would be considered in Fuzzy Systems as a membership function interpreted as "x_i is greater than 2.3/w_{ij}", where $f(2.3/w_{ij}) = 0.9$ (the function *f(x)* can reach 1 only asymptotically, thus we have set the α-cut for α=0.9) and, that the operation in (7), considering *f(* $x_i w_{ij}$ *)* as a membership function, represent the well-known Algebraic Sum operator, qualified as a S-norm (union), then a neural network can be mapped into a rule-based system. In the next section the process to extract rules from ANN will be presented.

C. *Extracting Rules from ANN*

From ANN shown in Figure 1, considering the hidden neurons without bias, $\sum_{i=1}^{n} x_i w_{ij} \geq 0$ and $x_i w_{ij} \geq 0$, for each neuron in hidden layer, one rule can be extracted as:

Rule R_j: If $\sum_{i=1}^{n} x_i w_{ij}$ is A then $y_j = b_j$ (8)

where A is a fuzzy set whose membership function is the positive-sigmoid function .

And, according to (7), rules as in (8) can be written as:

Rule R_j : If (x_1 w_{1j} is A) *…* (x_i $w_{i\,j}$ is A)* …* (x_n w_{nj} is A) **then** $y_j = b_j$ (9)

As the expression "x_i w_{ij} is A" might also be interpreted as "x_i is A_{ij} " (the fuzzy set A_{ij} has as membership function $m(A_{ij}) = f(x_i w_{ij})$, with the weight w_{ij} as a scaling of the slope of *f(.)*), and once the operation * is the Algebraic Sum operator (OR), let's rewrite (9) as:

Rule R$_j$: **If** (x_1 is A$_{1j}$) ***or…or*** (x_i is A$_{ij}$) ***or… or*** (x_n is A$_{nj}$) **then** $y_j=b_j$ (10)

where the rule's firing strength are calculated by the algebraic sum operator as follows:

$$v_j = m(A_{1j})*....*m(A_{nj}) =$$
$$1-((1-m(A_{1j}))...(1-m(A_{nj})) \quad (11)$$

Finally, from the output neuron in Figure 1, the output of the fuzzy system can be extracted as:

$$y=\sum_{j=1}^{m} b_j s_j \quad (12)$$

and since $s_j = v_j$ and $b_j = y_j$, the equation (12) can be rewritten in the following manner:

$$y=\sum_{j=1}^{m} y_j v_j \quad (13)$$

The inference system extracted previously from the neural net is similar to a zero-order Takagi-Sugeno model, with the difference that here the fuzzy logic operator used to calculate the firing strength of each rule is a S-norm (OR) and not a T-norm (AND).

However, for each S-norm there is a T-norm "associated" with it, where "associated" means that there exist a fuzzy complement such that the two together satisfy the DeMorgan's Law.

The T-norm associated with the Algebraic Sum operator $S(a,b)=1-(1-a)(1-b)$ is the Algebraic Product operator $T(a,b)=ab$.

Therefore, the extracted rule system in (10) can be transformed to:

Rule R$_j$: **If** (x_1 is Not A$_{1j}$) ***and…and*** (x_i is Not A$_{ij}$) ***and…and*** (x_n is Not A$_{nj}$) **then** $y_j=b_j$ (14)

where the firing strength for each R$_j$ rule is now calculated by the algebraic product operator (AND operator) and the system output is as follows:

$$y=\sum_{j=1}^{m} b_j(1-v_j) \quad (15)$$

Rearranging (15), the output of the fuzzy system is calculated by:

$$y=\sum_{j=1}^{m} b_j(1-v_j)=\sum_{j=1}^{m} b_j-\sum_{j=1}^{m} y_j v_j \quad (16)$$

where, $\sum_{j=1}^{m} b_j$ will be considered as the default value of the fuzzy system output.

If the bias is used in the hidden neuron then it will only change the consequent of the rule R_j of b_j to $b_j(1-f(q_j))$, and considering that the bias (q_{out}) of the output neuron is also used. The output of the system will be changed to:

$$y=\sum_{j=1}^{m} b_j-\sum_{j=1}^{m} y_j v_j+q_{out} \quad (17)$$

with $\sum_{j=1}^{m} b_j+q_{out}$ as the new default value of the rule.

D. *Comments*

The process explained so far contains the basic idea to produce the mapping of ANNs into FIS. However, for the rule antecedents extracted from ANNs to make sense - to be meaningful and subject to interpretation - we have the following condition:

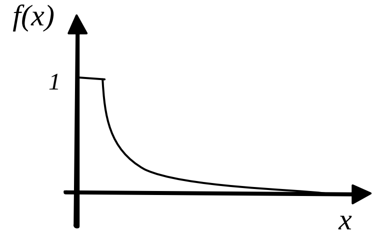

Fig. 4. New Membership Extracted

Condition 1: Considering (14), if the negation (NOT) is introduced in the extracted membership $m(A_i)=f(x_i w_{ij})$, we will have the new membership of the figure 4, and defined as:

$$f(x_i w_{ij})=\begin{cases} e^{-x_i w_{ij}} & , x_i w_{ij}\geq 0 \\ 1 & , x_i w_{ij}<0 \end{cases} \quad (18)$$

Considering the weight w_{ij} as the scaling factor of the *f(.)* and the α-cut for $a=0.999$, we can approximate (18) to :

$$f(x_i)=\begin{cases} e^{-x_i w_{ij}} & , x_i \geq 0.001/w_{ij} \\ 1 & , x_i < 0.001/w_{ij} \end{cases} \quad (19)$$

where $f\ (0.001/w_{ij})=0.999\approx 1$

The interpretation of this new set fuzzy is "smaller than $0.001/w_{ij}$"and it will only make sense if $0\leq 0.001/w_{ij}\leq 1$, which leads to $w_{ij}\geq 0.001$.

This consideration appears as a result from the usual practice of training an ANN with normalized inputs; therefore all memberships functions extracted have to be defined for the respective input interval.

With $w_{ij}\geq 0.001$, $0\leq x_i\leq 1$ and $q_j\geq 0$, the correct use of (7) is guaranteed since we will always have $\sum_{i=1}^{n} x_i w_{ij}\geq 0$ and $x_i w_{ij}\geq 0$.

However, during the training of the neural net the bias weights values can fall into the interval $[-\infty\ +\infty]$. To overcome this problem, in the next section we will present how to guarantee, during the learning process, the restrictions $w_{ij}\geq 0.001$ and $q_j\geq 0$ in such a way that we can extract rules from the neural network as presented later.

D. Constrained Neural Network

In section B we have seen that if we have $\sum_{i=1}^{n} x_i w_{ij}\geq 0$ and $x_i w_{ij}\geq 0$ we would extract rules as in (14), and in section C that, if $w_{ij}\geq 0.001$ and $q_j\geq 0$, we would always have extracted rules that may make sense.

Considering the restrictions $w_{ij}\geq 0.001$ and $q_j\geq 0$, let transform the weights and bias of the equation (1) using the exponential function:

$$s_j = f(\sum_{i=1}^{n} x_i(0.001+e^{w_{ij}})+e^{q_j}) \quad (20)$$

Using this transformation, the new weight $w'_{ij} = 0.001 + e^{w_{ij}}$ will be always greater than 0.001 and the new bias $q'_j = e^{q_j}$ will be greater than zero.

These transformations will not change the backpropagation algorithm commonly used for training the neural network. The algorithm will adjust normally w_{ij} and q_j between $[-\infty \;\; +\infty]$ and the restrictions will be guarantee through the exponentiation of w_{ij} andq_j.

E. Extraction of a Transparent Fuzzy System

The most important property of FIS, that distinguish them from other approaches such ANNs, is their capacity of explanation. In other words, FIS have the potential to express the behavior of real systems in a comprehensible manner. This property, known as transparency, enables the user to understand how each system parameter influences the output of the system.

Fuzzy transparency is directly associated to the concept of linguist interpretability. However, in fuzzy systems, transparency and interpretability are distinct terms. Interpretability is a property that exists by default being associated with linguistic rules and fuzzy sets, whereas transparency is the measure of how valid is the linguistic interpretation of the system and it is not a default property.

Although the methodology presented so far provided the extraction of an interpretable rule-based system, these rules cannot be considered transparent. In order to provide such transparency, an approximation process needs to be carried out on all membership extracted from the ANN. In this work, this process is performed by using a combination of 5 membership functions, which results in a new rule-based system with a total number of rules equal to 5^n, where n is the number of inputs of the system. Figure 5 shows the membership functions used in the approximation process.

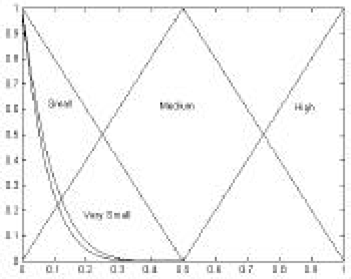

Fig. 5 – The new membership function for each input

E. A posteriori probabilities and Neural Network

ANNs have been successfully used in pattern classification task. They are inherently discriminative and yield estimates for a-posteriori probabilities of the classes when trained appropriately [6].

Let O$_1$ and O$_2$ be two populations of objects and, for $s \in \Omega_1 \cup \Omega_2$, let $x = x(s)$ be the m-dimensional feature vector of s.

The classical approach to linear discrimination between O$_1$ and O$_2$ is based on some linear function $g(x,s) = a^t x + b$ such that s is classified as coming from O$_1$ when $g(x,s) = a^t x + b > 0$ and from O$_2$ when $g(x,w) = a^t x + b < 0$. Linear discrimination can be also approached by considering the transformation:

$$p(\Omega_i / x) = f(a^t x + b) \qquad (21)$$

where f is the logistic function. In this case the linear discrimination is known in statistical literature as logistic discriminant function and (21) can be viewed as the a posteriori probability of class O$_i$ given a value of x.

Usually, in the case of only two classes, the threshold for the class decision is chosen to be equal to zero, which is the median of the standardized logistic function, then:

$$y_c = \begin{cases} \Omega_1 & ,\text{if } f(a^t + b) \geq 0.5 \\ \Omega_2 & ,\text{if } f(a^t + b) < 0.5 \end{cases} \qquad (22)$$

Using this rule for the decision, the value $f(a^t + b)$ can be view as:

$$p(\Omega_1 / x) = f(a^t x + b) \text{ when } f(a^t + b) \geq 0.5 \qquad (23)$$
$$p(\Omega_2 / x) = 1 - f(a^t x + b) \text{ when } f(a^t + b) < 0.5$$

where $p(\Omega_1 / x)$ is the a posteriori probability of class O$_1$ given a value of x and $p(\Omega_2 / x)$ is the a posteriori probability of class O$_2$ given a value of x.

In ANN domain, the logistic discriminant can be realized by the simple perceptron and can be also extended to multilayer neural networks. In this case, the a posteriori probability is calculated by the output neuron of the ANN, according to (2).

Considering the ANN as a logistic discriminant, all the process of the extraction of rules explained so far can be extended for the case when a logistic function is used in the output neuron. In this case, the extracted fuzzy system is considered as a linear discriminant. The Fuzzy System gives the classification for the pattern presented in the input according to:

$$y_c = \begin{cases} \Omega_1 & ,\text{if } y \geq 0 \\ \Omega_2 & ,\text{if } y < 0 \end{cases} \qquad (24)$$

where y is the output of the extracted fuzzy system calculated by (17). Afterwards, the a posteriori probability of class Q given a value of x is calculated using the logistic function:

$$p(\Omega_1 / x) = f(y) \quad \text{when } y \geq 0 \qquad (25)$$
$$p(\Omega_2 / x) = 1 - f(y) \text{ when } y < 0$$

The process is illustrated in Figure 6.

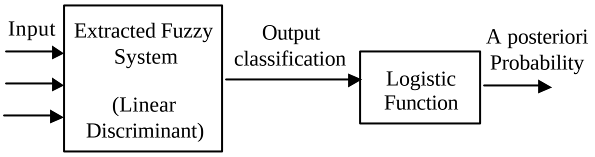

Fig.6. Calculating the a posteriori probability

IV. TRANSFORMER FAULT DIAGNOSIS

A. The proposed system for fault diagnosis

The detection of incipient faults on transformers follows, in general, the flow chart presented in figure 7.

The process begins with the observation of the evolution rate of combustible gases that exceed “normal” quantities. If the evolution rate per day is greater than a determined level then, the transformer may have an active internal fault. The possible fault is investigated by DGA methods.

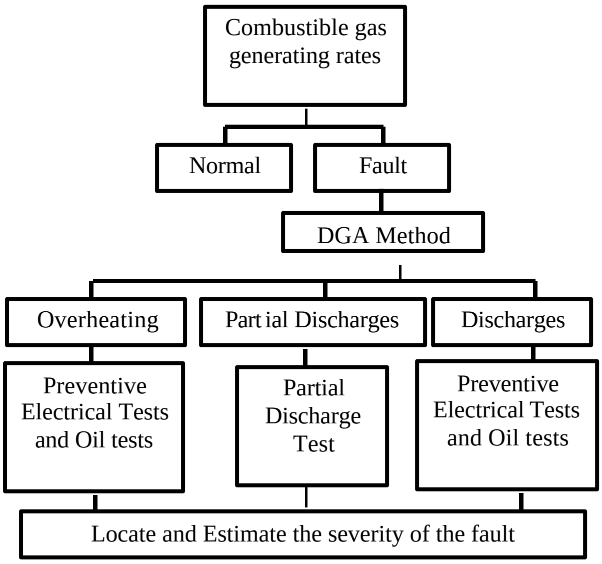
Combustible gas generating rates
Normal
Fault
DGA Method
Overheating
Part ial Discharges
Discharges
Preventive Electrical Tests and Oil tests
Partial Discharge Test
Preventive Electrical Tests and Oil tests
Locate and Estimate the severity of the fault

Fig.7. Fault Transformers Diagnosis Flow Chart

After detection of the possible fault, in order to obtain more detailed information, such as location of the fault, other tests are needed.

Many techniques for the detection of possible faults of transformer using the measurement of gases have been established. However, the search for a more reliable method using DGA is still a topic of interest in many utilities.

Some works have reported the use of ANN for transformer faults diagnosis. With Neural Network the fault diagnosis can be reduced to an association process of inputs (pattern gases concentration) and output (fault type) since it does not need a physical model. Neural networks are capable of acquiring experiences from training data and interpolate from it. However, for a proper training, the database has to be plentiful and consistent.

In this work, to take into account the ability of Neural Networks to automatically acquire experiences from training data, we propose its use for classification of transformer incipient fault.

The neural network trained to this task receives as input data the percentage of concentration of the gases methane (CH_4), ethylene (C_2H_4) and acetylene (C_2H_2) and then classifies the fault as discharges or thermal fault. As far as the classification system is concerned, only these tree gases were necessary as input for the ANN to give good results.

The database of faulty equipment inspected in service, used in Publication IEC 60599[7] and presented in [8], was used for training the ANN. Additionally, a database derived from the literature and also data obtained from CELPA (Power Stations of Pará, SA - Brazil) were also used in the training.

The study reported in this paper touches only the development of an ANN able to distinguish between thermal faults or discharges, not separating partial discharges . In a following publication the authors plan to report the behavior of a more complex diagnosis system taking in account all possible fault types.

The example is completed with the extraction of rules from the ANN, allowing a human specialist to understand how a particular diagnosis may have been generated

B. ANN Results

A neural network was trained for diagnosing between discharges or thermal faults. This ANN had 25 hidden neurons, 3 normalized inputs (gas concentration: C_2H_2, C_2H_4, and CH_4) and one output (0 - Thermal fault and 1 - discharge fault). We have used 383 training patterns and 115 testing patterns.

Table 1 shows the results of the ANN trained with the restrictions necessary for the rule extraction process, i.e., $w_{ij} \geq 0.001$ and $q_j \geq 0$. The result presented corresponds to best one after some training realized. The table also shows, for comparison, the results for the IEC method.

One may observe that the IEC method fails to identify a certain percentage of faults, but these have all been correctly classified by the ANN. Also, the IEC method is not exempt of error. Therefore, the ANN provided a more reliable classification of faults with a significant advantage.

Table 1 - Classification results

	TR %	***T1%***
ANN	100	93.91 (7 errors)
IEC 599	96.86 (1 error and 11 *NI**)	88.69 (13 *NI*)

TR% - percentage of correct diagnosis for the training set
T1% - percent of correct diagnosis for the test set
**** NI - not identified fault***

Given the ANN classifier, and given an input pattern, one may associate a probability to the classification by adopting the logistic function method described. Associated with the ANN output neuron, this gives the a posteriori probability for a certain fault in accordance to the decision rule presented in (24) and (25). Thermal faults are represented by O_1 while discharges are represented by O_2.

Table 2 shows an example for the result in two cases, where the classification (T or D) comes associated with a probability of correctness p.

Table 2 - Classification results

	H_2	*CH_4*	*C_2H_2*	*C_2H_4*	*C_2H_6*	*fault*	*p*
1	50	100	9	305	51	T	0.88
2	26	5.4	1.9	2.9	0.9	D	0.99

T -Thermal faults *p – a posteriori probability*
D – Discharges *Gas values in ppm*

C. Rule Extraction from the Neural Network

Once the ANN is trained, the process of extraction of rules from the ANN can be initiated.

The extracted FIS will have three normalized inputs (percentage of gases concentration: C_2H_2, C_2H_4, and CH_4) and one output (fault); as the trained ANN has 25 neurons in its hidden layer, 25 rules will be extracted. Each rule extracted will be expressed as:

R_i: *IF* (C_2H_2 is smaller than *a) AND* (C_2H_4 is smaller than b*) AND* (CH_4 is smaller than *c) THEN* $y_i = d$

The output of the Fuzzy system is calculated by (17), with $n = 25$. By using the logistic function, the posteriori probability can be calculated according to figure 6.

To guarantee the transparency of the fuzzy systems, all membership extracted from the ANN were approximated by the combination of the 5 membership functions showed in figure 5. With this combination, the number of the extracted rules will be now $5^3 = 125$, where each data input has 5 membership associated. The transformed rules will be now as:

R_1: *IF* (C_2H_2 is Small*) AND* (C_2H_4 Small*) AND* (CH_4 is small*) THEN* $y_1 = d$

R_2: *IF* (C_2H_2 is Small*) AND* (C_2H_4 Small*) AND* (CH_4 is Medium*) THEN* $y_2 = e$

R_3: *IF* (C_2H_2 is Small*) AND* (C_2H_4 Medium*) AND* (CH_4 is small*) THEN* $y_3 = f$

…

R_{125}: *IF* (C_2H_2 is High*) AND* (C_2H_4 High*) AND* (CH_4 is High*) THEN* $y_{125} = g$

V. CONCLUSION

In this paper, we have presented the derivation and definition of a transform that maps an ANN to a TS-FIS. We have explained how to proceed in a practical case and have demonstrated how the transform made explicit and put into light knowledge that was hidden in the ANN architecture.

This transformation methodology was inspired on the concept of f-duality that allowed to find the equivalent mathematical operation for a hidden neuron, which can be considered the foundation for all process of extraction of rules.

We have demonstrated that the process of extraction can be applied when the ANN output neurons have a linear function as well as when the logistic function is used. In the latter case, the extracted fuzzy system is considered a linear discriminant and the logistic function gives an a posteriori probability.

It is important to emphasize that any method of rule extraction from ANN is valuable only to the degree to which the extracted rules are meaningful and comprehensible to a human expert. In this work, we have presented an approximation process that guaranteed the transparency of the extracted rule-based system. This approximation, depending on the number of inputs of the system, can lead to a system with a considerable number of rules, which can affect the readability of the rule-base system. However, it seems that less readability is the price one has to pay for to guarantee the transparency of the system.

As far as a transformer fault diagnosis system is concerned, the results obtained with the ANN, as well as with the extracted fuzzy system, can be considered satisfactory.

The fuzzy system extracted will help the users to have more confidence in the fault diagnoses produced, giving a possibility to the specialist to interact more efficiently with the system, understand why a diagnosis is formed and perhaps learn some unexpected rules that nevertheless emerge from a system with a good classification performance.

VI. ACKNOWLEDGMENT

The authors are grateful to GRICES (Portugal) and CAPES (Brazil) for the support brought to the NESC/INESC cooperation protocol which allowed Ms. Castro to develop the research reported.

VII. REFERENCES

[1] Yann-Chang Huang. “Evolving Neural Nets for fault Diagnosis of Power Transformer”, IEEE Transactions on Power Delivery, Vol 18, Nº 3,pp 843-848, July, 2003.

[2] K. Tomsovic et al. “A Fuzzy Information Approach to Integrating different Transformer Diagnostic Methods”, IEEE Transactions on Power Delivery, Vol 8, Nº 3,pp 1638-1644, July, 1993.

[3] Y. Zhang et al, “ An Artificial Neural Approach to Transformer Fault Diagnosis”, IEEE Transactions on Power Delivery, Vol 11, Nº 4, pp 1836-1841, October,1996.

[4] Adriana Castro, Vladimiro Miranda, "Mapping Neural Networks Into Rule Sets and Making Their Hidden Knowledge Explicit – Application to Spatial Load Forecasting", *Proceedings of PSCC02 - 14th Power Systems Computation Conference*, Sevilla, Spain, June 2002

[5] J. M. Benitez et al, “Are Artificial Neural Networks Black Boxes?”, IEEE Transactions on Neural Networks, Vol 8, Nº 5, pp 1156-1164, September, 1997.

[6] M. D. Richard, & R. P. Lippmann, “Neural Network classifiers estimate Bayesian a posteriori probabilities”. Neural Computation, 3(4):461-483

[7] IEC Publication 60599, March 1999.

[8] M. Duval and A. Pablo, “ Interpretation of Gas-in-oil Analysis using new IEC Publication 60599 and IEC TC10 Databases”, IEEE Electrical Insulation Magazine, March/April, Vol17, Nº2, pp 31-41.