IMPROVED CONDITION ASSESSMENT BY FUZZY-MODELLING, ADJUSTMENT AND MERGING OF DGA'S INTERPRETATION METHODS

Markus Fischer[1*] and Stefan Tenbohlen[2]
[1, 2] Institute of Power Transmission and High Voltage Technology,
Pfaffenwaldring 47, 70569 Stuttgart, Germany
*Email: markus.fischer@ieh.uni-stuttgart.de

Abstract: Condition assessment is an important part of the dissolved gas analysis. The paper presents three essential strategies which aim at the improvement of the interpretation methods used for dissolved gas analysis. The strategies are: Fuzzy-modelling, adjustment and merging. Firstly, fuzzy-modelling is addressed. Interpretation methods are recreated by fuzzy inference systems. In doing so, thresholds used by the original methods are replaced by condition probabilities. Secondly, the paper discusses the strategy of adjustment. This strategy intends to adjust the recreated interpretation methods by training of the subjacent fuzzy inference systems with verified transformer conditions. The interpretation methods thereby improve their accuracy and reliability. Thirdly, the strategy of merging is presented. This strategy merges the outputs of all interpretation methods in a condition tree. The result is an interpretation method that identifies more transformer conditions than the particular interpretation methods. Furthermore, the method's accuracy and reliability is further improved. Finally, the paper discusses the results of the applied strategies.

1. INTRODUCTION

Power transformers play a crucial role in present-day risk based asset management of power grids. The outage of huge power transformers cause high maintenance expenses and presumably lead to an inadequate electrical power supply. In order to prevent power transformers from outages, different inspection techniques are in use. One approved technique is the dissolved gas analysis (DGA).

DGA, in turn, knows various methods to assess power transformer's condition, such as "General Electric", "Doernenburg Ratios" or "Duval Triangle" [1]. These interpretation methods show two drawbacks: At first, methods use thresholds either for key gases or for key gas ratios to decide on power transformer's condition. As a result, it is possible that methods decide on different conditions even at the time when vectors of key gases or key gas ratios are very similar. Secondly, most diagnostic methods do not allow for different conditions at once. But, as a matter of fact, power transformers can be in a complex condition that is a superposition of simple conditions [2]. In section two fuzzy inference systems (FIS) are introduced in order to solve these problems by fuzzy-modelling.

Furthermore, the accuracy of DGA's interpretation methods is improvable. Faults that are discovered by inspection of defective and therefore decommissioned power transformers are sometimes different to faults identified by DGA's interpretation methods. Moreover, DGA's interpretation methods sometimes state that a transformer is faulty when it is actually healthy. In order to avoid these problems, section three introduces a training approach for FIS. The training takes cases of verified faults and healthy conditions so as to adjust interpretation methods for a better hit ratio [3].

Finally, one can observe, that interpretation methods differ in case of their distinguishable conditions. While some decide rather roughly, others go into detail. In addition, the output of these interpretation methods can coincide or differ from case to case. In order to face these problems, section four introduces a merging strategy which merges different interpretation methods in a unified condition tree.

2. MODELLING WITH FUZZY INFERENCE SYSTEMS

DGA-based condition assessment of power transformers is very similar to condition diagnostic in medical science. In medical science medical scientist take blood samples first, secondly they analyse blood components qualitatively and quantitatively and finally they identify human's physical condition. DGA in turn uses oil samples in order to analyse dissolved gases qualitatively and quantitatively. Depending on the results, the power transformers' condition is assessed (Figure 1).

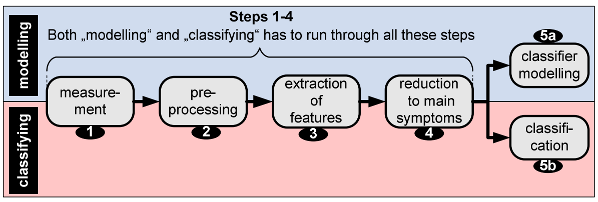

Figure 1: Five main steps for classifier modelling and condition classification.

The paper does not address sampling and measurement (Figure 1.1). Subject of investigation is the treatment of already measured gas values with the aim to improve the identification of conditions. On that account, Figure 1.2-5 depicts all steps that are addressed in this paper.

Figure 1.2 depicts pre-processing, which can be performed in order to improve the quality of gas values. Examples for pre-processing are: Averaging of gas values along the time or correction of systematic variations. Thirdly, gas values must be transformed into the feature space (Figure 1.3). Concretely, ratios of gas values have to be calculated and thresholds have to be applied to gas values and gas ratios. In Figure 1.4 the set of features is reduced to the characteristic symptoms of each condition type. In opposite to medical science, the authors use the term "symptom" not only for faulty conditions, but also for healthy and undefined conditions. Finally, the symptoms need to be mapped to the corresponding conditions. The mapping is performed by a classifier (Figure 1.5b). A classifier can be modelled based on one of many well known interpretation methods like for example "General Electric", "Doernenburg Ratios" or "Duval Triangle" (Figure 1.5a). Of course other classifiers are applicable. In particular those classifiers are applicable that work with fuzziness and probabilities.

2.1. Classifier's properties

Classifiers map features that are symptomatic for a condition as symptoms to the condition, while classifiers don't map features that are asymptomatic. It is important that classifiers are surjective. Surjectivity means: For each condition C there is at least one feature (F) which is mapped to the condition as a symptom (S). Classifiers, which are not surjective, cannot identify all conditions. Figure 2 depicts a surjective classifier by example of General Electric. The classifier maps symptom $S_{\text{thermal fault (high temperature)}}$ to the condition $C_{\text{thermal fault (high temperature)}}$, while it doesn't map the other symptoms to that condition. In order to achieve a clear view, the mappings for the other conditions are not depicted.

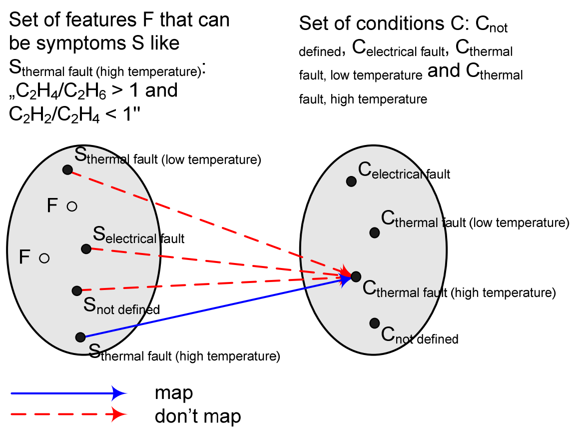

Figure 2: The surjective classifier of General Electric.

2.2. Fuzzy classifiers

In order to avoid thresholds and in order to state probabilities for transformer conditions it is necessary to build classifiers by means of FIS. It is neither the intention of the paper to present details about the functionality of FIS nor to show how FIS help to build fuzzy classifiers. An introduction to the basics of FIS is given in [4] and an introduction to the basics of classifier modelling is presented in [5]. Instead the paper focuses on the replacement of thresholds by condition probabilities.

The original classifier of General Electric, which is set-theoretic regarded in Figure 2, works with exact symptoms that use thresholds. Thus, the mapping is exact, too. Figure 3 presents a spatial view of the original General Electric classifier. As one can see, the exact symptom "$C_2H_4/C_2H_6 > 1$ and $C_2H_2/C_2H_4 < 1$" is mapped to thermal fault (high temperature), while all other symptoms aren't mapped to thermal fault (high temperature).

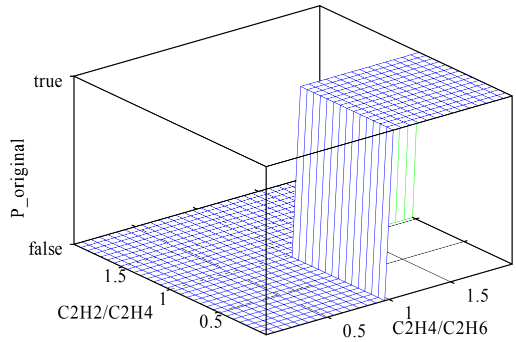

Figure 3: Original General Electric classifier that maps exact symptoms to "thermal fault (high temperature)".

FIS are able to make exact classifiers fuzzy. Set-theoretic spoken, fuzzy classifiers can be understood as gradual mapping of symptoms to transformer conditions. There is no "map to" or "don't map to" anymore. Mapping of symptoms to conditions with FIS is then a matter of degree (Figure 4).

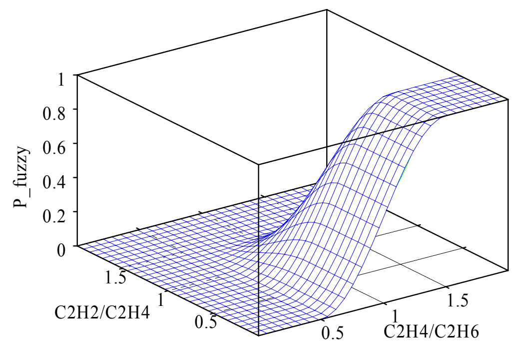

Figure 4: Fuzzy version of original Figure 3 classifier.

Fuzzy mapping is achieved by proper membership functions. The paper deals with membership functions derived from cosine. Thus, these membership functions are point symmetric and cover the whole range from 0 to 1. For the fuzzy classifier of General Electric, which is depicted in Figure 4, two types of membership functions are used: Firstly, the membership function of Equation 1 and secondly the complementary membership function of Equation 1.

$$f(x) = \begin{cases} 1, \text{for } x < 0.5 \\ (1+\cos((x-0.5)\cdot\pi))/2, \text{for } 0.5 \le x \le 1.5 \\ 0, \text{for } x > 1.5 \end{cases} \quad (1)$$

3. ADJUSTMENT OF INTERPRETATION METHODS BY TRAINING

Fuzzy classifiers do not inevitably identify transformer conditions perfectly. Verified transformer conditions associate with measured gas values can agree or disagree with classifier's results. It is the purpose of training to adjust classifiers where required.

The paper deals with training data from the IEC TC 10 database which is presented in [6]. The database provides training data for different equipment. In order to improve the classifier for power transformers with a "none-communicating" OLTC, the paper deals with the corresponding training data. If the classifier should be improved for other equipment, other data has to be used for training. In sum, 41 training data are used. In order to save space, Table 1 shows only a snippet of the training data. The training data consist of measured gas values and identified faults by inspection of the equipment. These training data is listed in excerpt in row 5-8 of Table 1. Furthermore, the database provides training data that consists of measured gas values where the equipment in 90% of all cases is identified as normal operating (healthy condition). In Table 1 these training data is listed in row 1-3.

Table 1: Snippet of training data taken from [6]

$\frac{C_2H_4}{C_2H_6}$	$\frac{C_2H_2}{C_2H_4}$	Not defined e.g. healthy	Electrical fault	Thermal fault (low temp.)	Thermal fault (high temp.)
1.00	0.10	90 %	0 %	0 %	0 %
2.34	0.06	90 %	0 %	0 %	0 %
0.77	0.30	90 %	0 %	0 %	0 %
…	…	…	…	…	…
12.64	0.04	0 %	0 %	0 %	100 %
6.05	0.03	0 %	0 %	0 %	100 %
2.81	0.00	0 %	0 %	0 %	100 %
…	…	…	…	…	…

Whilst the fuzzy classifier is trained with all 41 training data it changes its shape a little (Figure 5).

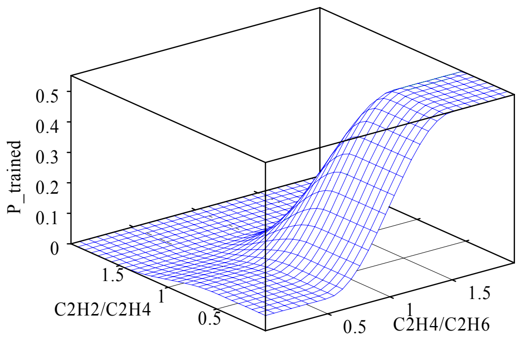

Figure 5: Trained fuzzy classifier of Figure 4.

The figure shows that the training only affects some areas. In order to make it easier to identify where changes took place, Figure 6 pictures the training impact solely by showing the difference of the trained fuzzy classifier (minuend) and the untrained fuzzy classifier (subtrahend).

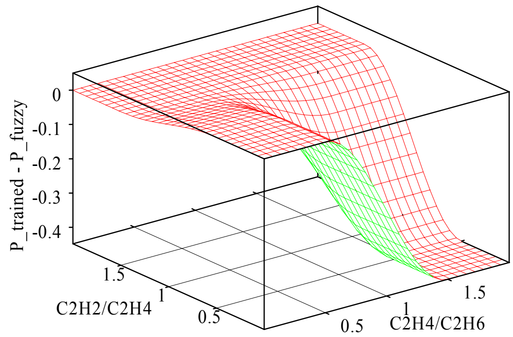

Figure 6: The impact of training to the untrained fuzzy classifier.

Regarding Figure 6, one notice two areas which are visibly (and actually) affected by training:

1. $C_2H_4/C_2H_6 > 1$ and $C_2H_2/C_2H_4 < 1$: The probability of thermal fault (high temperature) is notably decreased.

2. $C_2H_4/C_2H_6 < 1$ and $C_2H_2/C_2H_4 < 1$: The probability of thermal fault (low temperature) is slightly increased.

3.1. Training algorithm

The paragraph above is about training and training results on a user level. In order to validate the results of the training (trained fuzzy classifier), it is important to have a look at the training algorithm itself, which is given by Figure 7.

Beforehand, one should recall that classifiers are constructed by FIS. That is why the training algorithm has to operate on FIS level. The training itself is originally tied to artificial neural networks and not to FIS. If one focuses on a special category of artificial neural networks, namely radial basis function artificial neural networks, and a special category of FIS, namely Sugeno type FIS, one can observe that both are equivalent [7]. Hence, at least in this case it is possible to carry over the training feature of the artificial neural networks to the FIS.

Furthermore, it is important to know that FIS internally map each symptom to a condition by exactly one production rule and this production rule is weighted. The weight takes values in the range of [0, 1] to indicate the relative reliability of the corresponding production rule. In order to switch between a production rule and the equivalent number of training data, it is necessary to keep the trust factor. This trust factor cannot be held inside the FIS and should therefore be saved somewhere else.

```
01 load fis, trustFactor, trainingVectors
02 for each r of fis.rules {
03   r.weight = r.weight * trustFactor
04 }
05 for each t of trainingVectors {
06   for each r of fis.rules {
07     if r.premise(t.C2H2DivByC2H6, t.C2H2DivByC2H4) > 0,5 {
08       r.conclusion = (r.conclusion * r.weight +
09                       t.thermalFaultLowTemperature) / (r.weight + 1)
10       r.weight = r.weight +1
11     }
12   }
13 }
14 trustFactor = 0
15 for each r of fis.rules {
16   trustFactor = maximum(trustFactor, r.weight)
17 }
18 for each r of fis.rules {
19   r.weight = r.weight / trustFactor
20 }
21 save fis, trustFactor
```

Figure 7: Training algorithm written in pseudo code.

In line 1 the training algorithm loads the FIS, the trust factor and finally the training vectors. In the FIS all production rule weights are initially set to 1. That means the relative reliability of all production rules are taken to be equal. The trust factor is initially set to 100. It indicates that the most reliable production rule is as meaningful as 100 training data. In subsequent lines 2-4 the training algorithm de-normalises production rule weights. Afterwards in lines 5-13 the training algorithm identifies for each training vector which production rule has to be modified. Then the conclusion and weight of the identified production rule are modified. The new conclusion is defined as the weighted average of the old conclusion and the statement of the training data. The new weight is defined as the old weight increased by one. After that, the new trust factor is calculated in lines 14-17. Then de-normalisation of production rule weights take place in lines 18-20. Finally, the algorithm saves the modified FIS and the new calculated trust factor in line 21.

Summing up, one can say that the training algorithm calculates the arithmetic mean to train the FIS according to the training data. The relative reliability of a production rule is represented by its weight. The trust factor is needed to compare production rules and training data.

3.2. Regression analysis and expectation value

In the paragraph above the approach of weighted average is used in order to train the FIS. But the question is still unanswered, whether the approach is meaningful or not.

Training of fuzzy classifiers is in fact the same as curve fitting. In the context of statistics curve fitting is called regression analysis. The aim of regression analysis is to fit a given parameterised function to given data points. Typical parameterised functions are polynomial like in equation 2. Equation 2 is a parameterised polynomial function with two variables, x_1 and x_2. Thus, it is capable to fit data points by a polynomial surface.

$$f_1(x_1, x_2) = a_0 + b_1 x_1 + b_2 x_2 + c_1 x_1^2 + c_2 x_2^2 + \ldots \quad (2)$$

In order to adjust a classifier's surface there are two fundamental strategies: Either fitting the whole classifier at once globally or fitting the classifier one by one locally. The paper deals with local fitting. That is, because classifiers are constructed by FIS, FIS in turn use locally mapping production rules and the purpose of training is to adjust these production rules. Furthermore, changing the shape of membership functions is not the objective. Only levels are of interest. For that reasons the equation 2 is reduced to equation 3.

$$f_1'(x_1, x_2) = a_0 \quad (3)$$

The aim of regression analysis is to define the undefined parameter of the parameterised function, which in case of Equation 3 is only a_0. For parameter definition, regression analysis most often uses the minimum square error method (equation 4).

$$f_2(a_0) = (p_1 - a_0)^2 + (p_2 - a_0)^2 + \ldots + (p_n - a_0)^2 \quad (4)$$

The least square error method is derived from the Euclidian norm, which is the most descriptive norm at all. In order to define a_0, the Equation 4 has to be differentiated first. Secondly, the derivate must be set to zero and then finally solved for a_0. The result of the calculation is given by equation 5.

$$a_0 = \frac{p_1 + p_2 + \ldots + p_n}{n} = \frac{\overbrace{(n-1) \cdot p}^{\text{if } p = p_1 = p_2 = \ldots = p_{n-1}} + p_n}{n} \quad (5)$$

Obviously the first part of equation 5 defines a_0 as the arithmetic mean of all p_i. The arithmetic mean in turn is an estimating function for the expectation value E(P). P is the random variable of the statements of all training data p_i, where p_i is either "thermal fault (high temperature)" or not "thermal fault (high temperature)". The law of large numbers says: The more training data, the better the estimation of the expectation value. The second part of equation 5 is defined as the weighted arithmetic mean. The first and the second part are equal, when p is equal to p_1, p_1,…, p_{n-1}. The second part is used in Figure 7, lines 8-9, where the training algorithm iteratively calculates the new conclusion of a production rule.

4. MERGING METHODS IN A CONDITON TREE

Previous paragraphs give hints to improve a classifier of a certain interpretation method. Each interpretation method uses an individual classifier to state the probability of a couple of conditions. Among all conditions there are conditions which can be identified by all classifiers and some conditions which can be only identified by a subset of classifiers. It is up to this paragraph to show how these classifiers can be joined.

In the first instance, each classifier needs to have a separate condition tree. Keeping the example of General Electric, its condition tree is exposed by Figure 8.

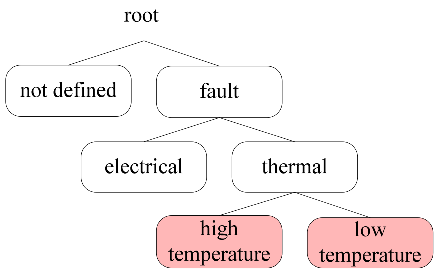

Figure 8: Condition tree of General Electric.

It is a hierarchical condition tree, where each node holds twice: At first it holds the probability (P) of a certain transformer condition. At second it holds the reliability (R), which indicates the reliability of the probability estimation. General Electric's classifier itself identifies only a subset of these conditions, namely "not defined", electrical fault", "high temperature" and "low temperature". For these conditions the probability is given by the classifier and the reliability is given by the product of the weight of the most important production rule and the trust factor. Thus, it is necessary to calculate the possibilities and reliabilities of the remaining nodes.

Among all successor nodes of a node, the reliability of the node is defined as the reliability (R) of the successor node ($Succ_i$) with the highest probability (Max_P) (Equation 6). Meanwhile among all successor nodes of a node the probability (P) of the node is taken as the probability of the successor node with the highest probability (Equation 7). The algorithm that calculates each node's probability and reliability runs bottom up, beginning with the lowermost nodes and ending with the root.

$$R(node) = R(Max_P(Succ_1(node), \ldots, Succ_n(node))) \quad (6)$$

$$P(node) = P(Max_P(Succ_1(node), \ldots, Succ_n(node))) \quad (7)$$

At least one additional classifier is needed to merge classifiers in an entire condition tree. The paper deals with the classifier of Doernenburg Ratios. That is to save space, because the related condition tree (Figure 9) is rather compact as it is the condition tree of General Electric (Figure 8). Again, the classifier is only able to identify a subset of conditions, namely "not defined", "partial discharge", "discharge" and "thermal". All remaining nodes must be calculated according to the procedure mentioned for the condition tree of General Electric.

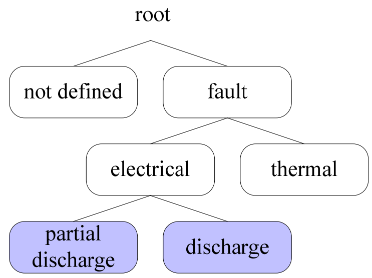

Figure 9: Condition tree of Doernenburg Ratios.

Figure 10 shows the entire condition tree. It spans its both parental trees, namely the condition trees of "General Electric' and "Doernenburg Ratios". Thus, it is self-evident that all its nodes have a corresponding node in at least one of the parental trees. While "partial discharge" and "discharge" or "high temperature" and "low temperature", respectively, have corresponding nodes in just one parental tree, all the other nodes have corresponding nodes in both parental trees.

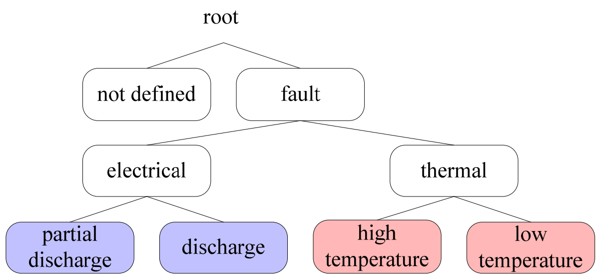

Figure 10: Entire condition tree spanning General Electric's and Doernernburg Ratios' condition tree.

For the entire condition tree, each node's reliability (R) is defined as the sum of the reliabilities of the corresponding parental nodes (Par_i) (Equation 8). Each node's probability (P) is defined as the weighted possibilities of the corresponding parental nodes (Equation 9).

$$R(node) = R(Par_1(node)) + \ldots + R(Par_m(node)) \quad (8)$$

$$P(node) = \frac{R(Par_1(node)) \cdot P(Par_1(node)) + \ldots}{R(Par_1(node)) + \ldots + R(Par_m(node))} \frac{\ldots + R(Par_m(node)) \cdot P(Par_m(node))}{R(Par_1(node)) + \ldots + R(Par_m(node))} \quad (9)$$

5. APPLICATION OF THE NEW METHOD

In order to show the performance of the introduced strategies, a real-life measurement is given by example. The example is about a defective 385MVA generator transformer. The fault is identified as partly broken conductor by inspection. The impacts are hot spot and partial discharge, whereby partial discharge touches cellulose isolation. The transformer oil contained the following gases in parts per million: hydrogen (300), methane (1740), acetylene (< 1), ethylene (3500) and ethane (1190). Initially, these gases are evaluated by the original classifiers of General Electric and

Doernenburg Ratios. Moreover, these gases are interpreted by the combined trained fuzzy classifiers of General Electric and Doernenburg Ratios (Figure 11).

The original General Electric classifier states "thermal fault (high temperature)". Doernenburg Ratios in turn detects the more general fault "thermal fault". According to that, both classifiers only identify the thermal part (hot spot) of the complex real-world fault (hot spot, partial discharge and degradation of cellulose). If one look at Figure 11, the condition tree shows a more diversified condition view. Starting with the first floor, just after root, the transformer condition is most probably identified as "fault" (P=64%), while there is a chance to have an "undefined condition" (P=31%) as well. On the second floor the fault most likely appears as "thermal fault" (P=64%), while there is a smaller portion of "electrical fault" (P=10%). On the third floor "thermal fault" branches in "thermal fault (low temperature)" (P=1%) and "thermal fault (high temperature)", which is the most likely fault (P=55%). The "electrical fault" cannot be further specified. To sum up, the condition tree states that the most dominant fault is "thermal fault (high temperature)", but in addition it states an "electrical fault", which is less dominant. The participation of cellulose cannot be stated, because this condition is not represented in the condition tree.

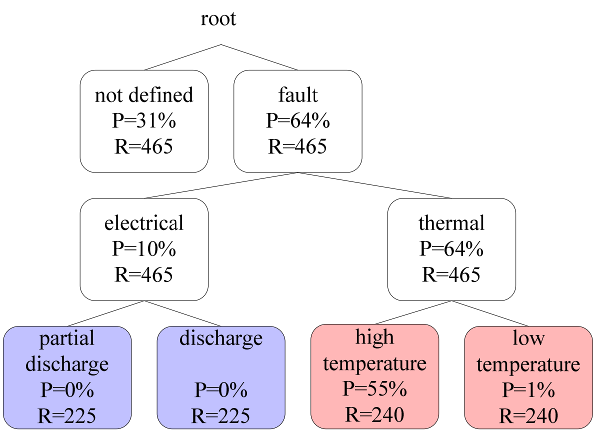

Figure 11: Entire condition tree, where each node's probability and reliability is identified.

In addition, each condition's probability is rated by the reliability R, which is the amount of training data supporting the probability. While the detailed conditions on the third floor are only identified by one parental tree (lower reliabilities), the more general conditions of the first and second floor are identified by both parental trees (higher reliabilities). In summary it can be stated, the more detailed conditions, the less reliable and the other way around.

6. CONCLUSIONS

At first the strategy of fuzzy-modelling was applied. Therefore classifiers of diagnostic methods are re-modelled by FIS. Fuzzy classifiers avoid the identification of completely different conditions in case of similar gas values. It therefore replaces thresholds by condition probabilities. As a consequence of this, fuzzy classifiers are able to handle complex faults.

Secondly, the adjustment-strategy was applied. It uses a training technique in order to further improve the classifier's accuracy and reliability. The training method implements the arithmetic mean algorithm. It is proven that the more (accurate) training data is used, the better the classifier approximates the ideal expectancy value classifier. However the arithmetic mean training is somehow sensitive to the so called runaways. It is worth a try to implement a median value training algorithm. It would be less sensitive to runaways and could also approximate the expectancy value classifier.

The final strategy merged trained classifiers in order to extend the amount of identifiable conditions and in order to improve the accuracy and reliability of condition identification. Conditions were hierarchically ordered according to their levels of detail. The more specific the condition, the less reliable is the estimation of its probability and the other way around. So it is up to the user if he prefers details or reliability.

7. REFERENCES

[1] J.-P. Gibeault, "Tutorial: On-line monitoring of key fault gases in transformer oil – operational experiences accumulated over the years", 10th ISH, Montréal, Canada, August, 1997

[2] J. Aragón-Patil, M. Fischer and S. Tenbohlen, "Improvement of dissolved gas analyses (DGA) by means of experimental investigation of generated fault gases and a fuzzy logic based interpretation scheme", 15th ISH, Ljubljana, Slovenia, August 27-31, 2007, Proceedings of the ISH, paper No. T7-157

[3] J. Aragon-Patil, M. Fischer, S. Tenbohlen; Z. D. Wang and M. Schäfer, "Investigation on Sampling, Measurement and Interpretation of Gas-in-Oil Analysis for Power Transformers", CIGRE Session 2008, Paper D1-204, Paris, France.

[4] T. Terano, K. Asai, M. Sugeno, "Fuzzy Systems Theory and it's Applications", Academic Press, San Diego, USA, 1992

[5] J. Aragon-Patil, M. Fischer, S. Tenbohlen, „Improvement of interpretation of dissolved gas analysis for power transformers", 3rd APTADM, Wroclaw, Poland, September 26-28, 2007, page No. 91-97

[6] M. Duval, A. dePablo, "Interpretation of Gas-In-Oil Analysis Using New IEC Publication 60599 and IEC TC 10 Databases", IEEE Electrical Insulation Magazine, Vol. 17, Issue 2, Mar./Apr. 2001, pp. 34-37

[7] H. C. Anderson, A. Lotfi, L. C. Westphal, "Comments on 'Functional Equivalence Between Radial Basis Function Networks and Fuzzy Inference Systems'", IEEE Transactions on Neural Networks, Vol. 9, No. 6, Nov. 1998, pp. 1529-1531