Flexible hierarchical neuro-fuzzy system for prognosis

FLEXIBLE HIERARCHICAL NEURO-FUZZY SYSTEM FOR PROGNOSIS

Erkki Jantunen
Technical Research Centre of Finland
Manufacturing Technology
P.O.Box 1702
FIN-02044 VTT, Finland
Erkki.Jantunen@vtt.fi

Abstract: An easy to configure hierarchical neuro-fuzzy system has been defined for the configuration of a prognosis system for condition monitoring of machinery. The system consists of a number of modules: data acquisition, signal processing, data handling, fuzzy classifier and a neural net for diagnosis. Data acquisition is based on the use of an AD card, and signal processing on the use of traditional FFT. The fuzzy classifier together with the neural network is organised in a hierarchical structure, which enables the easy configuration of the whole system. The approach is especially flexible in the sense that the total number of parameters the system can handle is not limited in practice. In the hierarchical structure the individual sub-models are restricted to handling eight fuzzy inputs simultaneously. In the system, the type of neural networks can be chosen from a list of choices based on the desired type of behaviour. In a normal case, parts of the hierarchical system are configured based on crisp information. Similarly, the features of neural nets are not used in all of the sub-models and they can be substituted with arithmetic expressions if there is no need for handling non-linear information or the behaviour is well known and can be easily defined otherwise.

Keywords: Condition monitoring, Diagnosis, Fuzzy logic, Neural networks, Prognosis, Signal processing

Introduction: Condition monitoring of rotating machinery has become increasingly popular in recent years as a result of better understanding of the financial values involved [1]. However, when organising condition monitoring in the plant environment, even though the transducers are still not cheap and cabling can be even more expensive, the problem in practice is the amount of work often involved in analysing the monitoring data. The analysis work is also very demanding, and it takes time to train people to a sufficient level of experience so that analysts become real professionals. Basically for the above reasons, quite a number of attempts have been made to automate the whole analysis and diagnosis procedure. The first kind of automatic analysis tools were rule-based expert systems. The rule-based approach as such can be considered in principle to be rather generic, assuming the developers have taken into account all possible situations which can occur with the machinery in question. However, herein already lies the problem in practice: only well-defined situations can be handled, and this in turn pushes the solution towards working only with very simplistic machinery. One way to overcome

this restriction is to use so-called case-based reasoning, where the principle is to develop a system which can document all possible problems or cases and the corresponding information from the transducers. Assuming that suitable information from analysis is available, this kind of approach should lead to quite a reliable result if there is time and resources to do the definition work. Since the features of condition monitoring signals can be rather complicated to analyse, and it is not always easy to know when a fault is present, different types of neural networks or statistical approaches have been used for this classification task. Basically the idea with the use of neural nets and other numerical methods is usually rather simple, i.e. let the net see a sufficient number of cases and it can then learn how the measured parameters are linked and consequently learn how faults can be recognised. Again, it is rather easy to say where the problem lies, i.e. how the system can be fed enough information from a set of transducers so that the whole range of interesting faults are covered in a remarkable set of running conditions. It is not the purpose of this introduction to try to cover the wide field of artificial intelligence (AI) and of knowledge based approaches to diagnosis of condition monitoring signals. Instead the idea is merely to show, using some examples, how solving one problem might lead to a range of other problems. There are so many approaches and none of them, although they work well in certain cases, are suitable for every kind of purpose, and that is why there is still room for new ideas and attempts in this demanding field of engineering and maintenance. The approach described in the following could be described as an attempt to combine a number of techniques referred to above in the most suitable way that would make the system easy to use, reliable, and wide in scope.

Principles of the approach: The system consists of a number of modules: data acquisition, signal processing, data handling, a fuzzy classifier and a neural net for diagnosis. Data acquisition is based on the use of an AD card, which can be configured to work with a number of sensors including, for example, vibration transducers. The system can also handle the on/off type of crisp information. Signal processing is mainly based on the use of traditional FFT (Fast Fourier Transform) together with ordinary statistical parameters. The novelty in signal processing and data handling lies in the use of regression analysis functions which make it possible to monitor a great number of different kinds of components, e.g. the tools in a machining process, without running into problems with available computer hard disk space. In the approach fuzzy logic, neural networks and case based reasoning are combined to build a system where the user can easily, through a graphical user interface, use and configure the system. The fuzzy classifier together with the neural network is organised in a hierarchical structure, which enables the easy configuration of the whole system. The approach is especially flexible in the sense that the total number of parameters the system can handle is not really limited in practice. In the hierarchical structure the individual sub-models are restricted to handling eight fuzzy inputs simultaneously (see Figure 1). The user can construct the whole diagnosis model through a graphical user interface. In practice, the most time consuming task is not the configuration of the system but the adjustment of the limits of the fuzzy classes, which again takes place through an easy to use graphical user interface with built-in editing features, such as copying. In the system the type of neural networks can be chosen from a list of choices based on the desired type of behaviour. In a normal case, parts of the hierarchical system are configured based on crisp information, and in

these sub-models the fuzzy classifier does not have its normal function but is merely used as such to render the treatment of data similar in all cases. Similarly, the features of neural nets are not used in all of the sub-models and they can be substituted with arithmetic functions or expressions if there is no need for handling non-linear information or the behaviour is well known and can be easily defined otherwise. The system has been programmed using Visual Basic programming language in a Windows operating system environment and is based on the use of multiple windows [2]. The major advantages in the proposed approach are its flexibility of working with different types of machinery and the possibility to copy parts of the model (=sub-models) from one industrial plant to another where similar components are used.
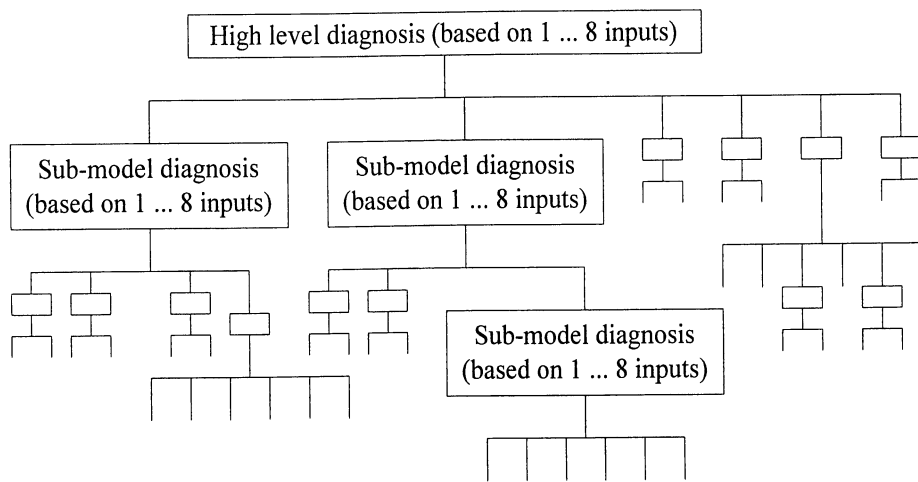

Figure 1. The structure of the hierarchical neuro-fuzzy system.

Data acquisition: Data acquisition is based on the use of an AD card, which can be configured to work with a number of sensors including, for example, vibration, sound, acoustic emission, pressure, current, voltage, power, speed of rotation and strain. The AD card is configured using a graphical user interface. The user is expected to define such parameters as the sampling rate, number of channels, type of windowing function, amplification/sensitivity, name of sensor, type (vibration, pressure, strain etc.) of sensor, units definition, type of averaging and number of averages. The idea is for the system to be capable of supporting a number of AD cards from a number of manufactures, although to date it has been configured to support only two models from different manufacturers.

Signal analysis: Signal processing is based on the use of traditional (spectrum, cepstrum) FFT (Fast Fourier Transform) together with statistical (root mean square, average, maximum, minimum, skewness and kurtosis) parameters and also the on/off type of information [3]. The user has a choice of these parameters and can assign from one to

eight of them to the system at the lowest defined classification level in a suitable optimal mix. The novelty in signal processing and data handling lies in the use of regression analysis functions which make it possible to monitor a great number of different kinds of components, e.g. the tools in a machining process [4]. When using regression analysis techniques, only the regression analysis coefficients are stored in the database. This markedly reduces the amount of data to be stored, especially if the system is used to monitor such a complex target that this might become a problem. The available regression functions are first-, second- and third-order polynomials, and a logarithmic function developed to indicate or follow the progress of wear and thus to be suitable for prognosis of the remaining lifetime of the machine component [5].

Database: All data used by the system is stored in an Access database. The neuro-fuzzy diagnosis part of the database consists of five tables, as shown in Table I along with the function of each. The database, although very easily described, is actually the key element of the whole system. All communication internally is through this database, i.e. the definition of the structure of the system is there, as is all the measured data saved there, taken from there for diagnosis, and the results of the diagnosis. The consequence of the above is naturally that the database size can with time become immense, although both signal analysis techniques and regression analysis are used to reduce the amount of data.

Table I. Tables of the database of the neuro-fuzzy diagnosis module.

Table	Description
Hierarchy	Describes the hierarchy of the system
Measurement-Data-Fuzzy	Gives the measurement results
Measurement-Conditions	Describes the measurement conditions
Text	Texts that the program uses for communication in different languages
Diagnosis	The results of fuzzy classification

Fuzzy classifier: In the approach, fuzzy logic, neural nets and case-based reasoning are combined to build a system which the user can easily configure through a graphical user interface. The fuzzy classifier, together with the neural network, is organised in a hierarchical structure which enables easy configuration of the whole system. The fuzzy classifier acts as a pre-processor to the neural net [6]. The approach is especially flexible in the sense that the total number of parameters the system can handle is not limited in practice (i.e. with the limitations given below there can be a total of 4681 lines in the

hierarchy table). In the hierarchical structure the individual sub-models are restricted to handling eight fuzzy inputs simultaneously. The number of hierarchy levels in the system is limited to four. The user can construct the model through a graphical user interface. In practice the most time-consuming task is not configuring the system, but adjusting the limits of the fuzzy classes, which again takes place through an easy-to-use graphical user interface show in Figure 2.
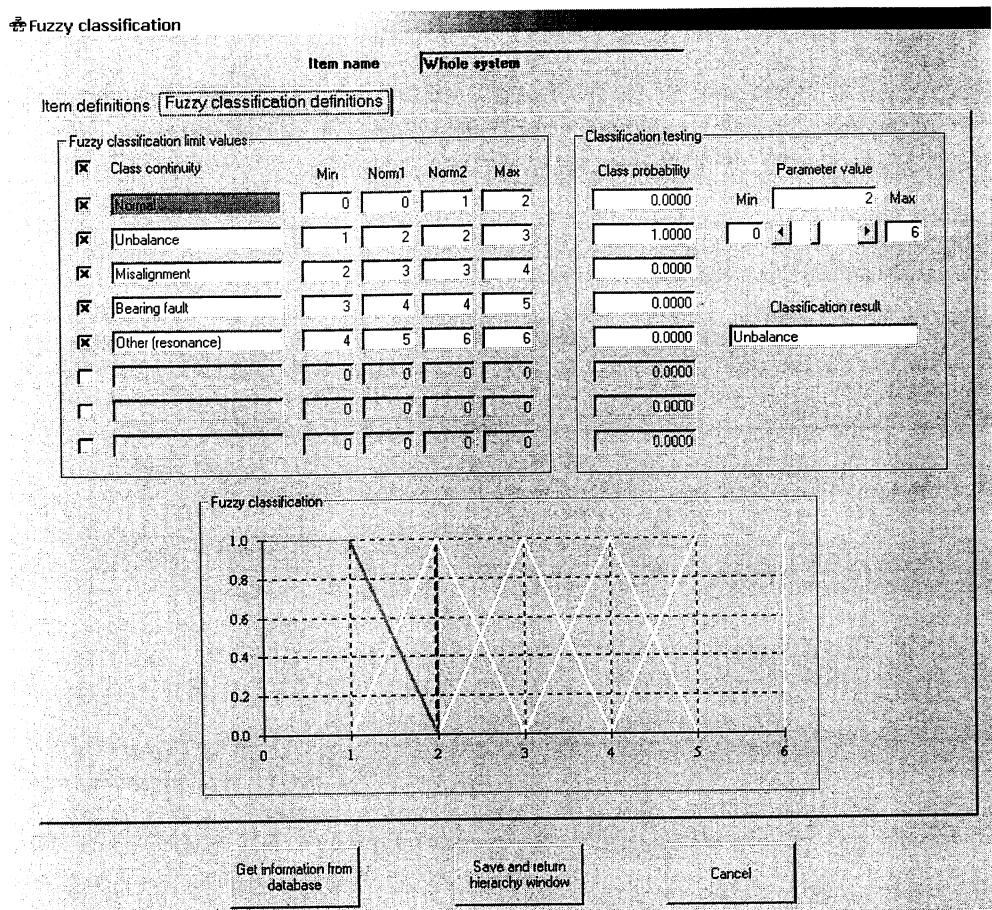

Figure 2. User interface of the fuzzy classification module.

The number of fuzzy classes can vary from one to eight. The classes must be continuous but the user can turn off checking for continuity while changes are being made, and then turn it on again. In the example shown in Figure 2, the number of fuzzy classes is five. For all of these inputs the user is expected to give four values which define the limits of that specific fuzzy class. However, two of these values actually define two values for the next class, i.e. only two additional values are needed for definition of the next class. In

practice this means that two plus n times two values are needed in the definition of n classes. The user interface shows the fuzzy classes graphically. It also gives the user the opportunity to test what happens when the parameter being classified gets a certain value, i.e. into which class it falls. It should be noted that all the measured parameters go through the fuzzy interface, and similarly all the results of fuzzy classification of lower sub-models pass through this interface. However, not all the classes need to be fuzzy, i.e. it is possible to define sharp limits between the classes. Sharp limits are often used when the results of fuzzy classification are passed further on, or when on/off type information is being handled. The example shown in Figure 2 is from a higher level, i.e. it is not the lowest level that handles parameters from condition monitoring signals or process status information. The example shows the fuzzy interface at the level where the diagnosis system distinguishes between a number of typical faults that can be diagnosed with the use of vibration measurements.

Diagnosis: In the system, the type of neural networks can be chosen from a list of choices based on the desired type of behaviour. In a normal case parts of the hierarchical system are configured based on crisp information, and in these sub-models the fuzzy classifier does not have its normal function but is merely used as such to make the treatment of data similar in all cases. Similarly, the features of neural nets are not used in all of the sub-models, and they can be substituted with arithmetic functions if there is no need to handle non-linear information. If the behaviour is well known, i.e. it is implicitly defined what combination of results a fuzzy sub-model means, this information can be defined into the system through the interface shown in Figure 3. When the updating routines of the system are started, the hierarchy is first optimised during which all unnecessary nodes of the hierarchy tree are deleted. After optimisation the system goes through all the nodes and levels, starting at the bottom. If classification between the levels is based on neural nets or other algorithms, the whole classification process proceeds automatically. However, if the user has chosen to specify that a certain combination in a sub-model should be translated or classified to correspond to a specific situation, i.e. to a certain number, this combination may not yet have been defined. Should this be the case, the system will stop and ask the user to make this specification. In the case of neural nets there is some variation depending on the type of nets used. In the case of a traditional feed forward network, it is assumed that the user will train the sub-model first so that it can handle all possible situations [7]. In the case of self-organised maps it is possible to let the system organise itself, so that after a learning period it can handle various situations. The aim is especially to configure a specific version of the QSOM routine [8] so it can be used as a self-organising map. For each of the sub-models, the system shows on the interface the corresponding interpretation of that model using a colour code. It also shows the result both as a number and as plain text if the cursor is moved to that point on the interface (see Figure 4). The idea is that when the system is running continuously, the user can easily identify where the indication of a fault or something peculiar appears in the system. More specifically, the system shows the item the user is looking at, and gives information about the fault. Naturally all this information has had to be defined for the system, and if the number of connected channels is high this might be quite a task. However, to make the system definition more

effective in the case of complicated systems, it is possible to copy information from one parameter to another.

☷ Interpretation of classification

Item name |Whole system   Hierarchy path :   **0**

┌─ Used channels and resulst classes ─────────────

| Item   Result class
Channel 1 |Tulk_Vär_Rad (mm/s) |1: Normal
             | |
Channel 3 |Tulk_Vär_Aks (mm/s) |2: Unbalance
             | |
             | |
             | |
             | |
             | |

┌─ Used classifier ─────────   ┌─ Result class ─────────
  |Fuzzy classification combination ▼| | ▼|

         Save and return   Cancel
         hierarchy window

Figure 3. The interpretation of fuzzy classification.

The copy and paste technique is very practical and saves a lot of time if the machinery to be monitored has, for example, a number of rolling bearings that are monitored using a number of acceleration sensors. In principle, all of these are monitored using basically the same set-up and fuzzy limits at the start, so it is easy to copy the definition of bearing monitoring for all of these bearings. In practice, the way these bearings behave may vary, which affects what sort of parameters should be used and what the exact fuzzy limits are.

However, it is a lot easier to do a little fine tuning than to define the same thing a number of times from scratch.
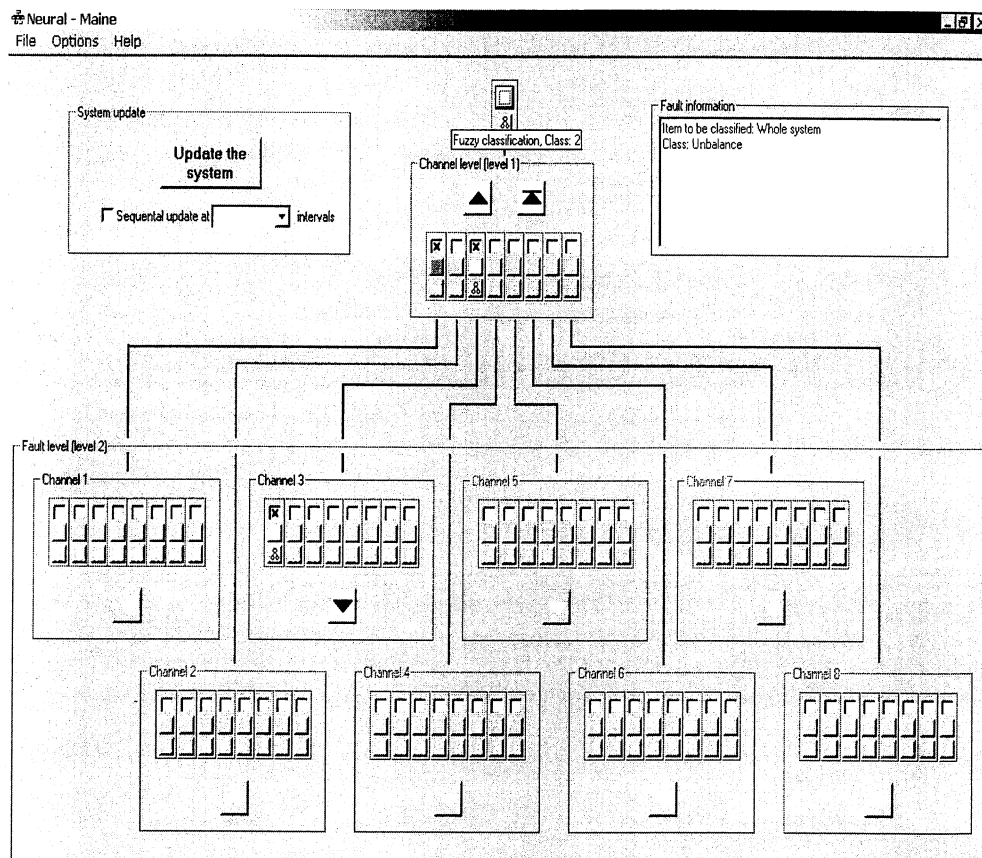

Figure 4. User interface of the hierarchical neuro-fuzzy prognosis system.

The time it takes for the system to go through all the sub-models with all parameters naturally depends on the number of parameters defined, and on the power of the computer. With a typical Pentium-type PC it takes only a few seconds if only a few channels are connected, or several minutes if a number of sub-models are connected. The user interface shows how the system is progressing. Naturally, if these times are compared with off-line monitoring lasting 2 weeks they are ridiculously short, but to a hurried user used to quick responses with a PC, they may feel a lot longer. Especially during the training and definition phase it may be frustrating to wait for the system to update, but it is possible to concentrate only on sub-models of interest by clicking off those parts that are not of interest. Even though a channel is turned off, the definition for that channel will be held in the database unless purposely altered or deleted. Because the

structure of the whole system is large it is impossible to show it all at once; consequently the user can move around in the model using the arrows shown in Figure 4. The user interface shows the level and channel in question.

Further development: The system is undergoing a testing phase in e.g. the food and manufacturing industries, power production and the monitoring of conveyors and lifts. The neural network part of the system cannot be regarded as complete due to the number of approaches that could be installed. Connections to different kinds of measuring equipment could widen the scope of the system. In many cases it would be logical to use the system on the World Wide Web as this could lower the cost of ownership, and offer ease of upgrading and a larger number of sub-models in the library [9]. Naturally the most important thing is to take into account feedback from industrial users, especially concerning any bugs in the system, and their views on how to make the system easier, faster and more logical to use.

Conclusion: An easy to configure hierarchical neuro-fuzzy system has been defined for the configuration of a prognosis system for condition monitoring of machinery. The system consists of a number of modules: data acquisition, signal processing, data handling, a fuzzy classifier and neural networks for diagnosis. Data acquisition is based on the use of an AD card, which can be configured to work with a number of sensors including, for example, vibration, sound and pressure. The system can also handle the on/off type of crisp information. Signal processing is based on the use of traditional FFT (Fast Fourier Transform) together with statistical time domain parameters. The novelty in signal processing and data handling lies in the use of regression analysis functions which make it possible to monitor a great number of different kinds of components, like the tools in a machining process. The fuzzy classifier together with the neural network is organised in a hierarchical structure, which enables easy configuration of the whole system. The approach is especially flexible in the sense that the total number of parameters the system can handle is not limited in practice. In the hierarchical structure the individual sub-models are restricted to handling eight fuzzy inputs simultaneously. The user can construct the model through a graphical user interface. In practice the most time-consuming task is not the configuration of the system but the adjustment of the limits of the fuzzy classes, which again takes place through an easy-to-use graphical user interface. In the system the type of neural networks can be chosen from a list of choices based on the desired type of behaviour. In a normal case parts of the hierarchical system are configured based on crisp information, and in these sub-models the fuzzy classifier does not have its normal function but is merely used as such to render the treatment of data similar in all cases. Similarly, the features of neural nets are not used in all of the sub-models and they can be substituted with arithmetic functions if there is no need for handling non-linear information, or if the behaviour is well known and can be easily defined otherwise. The major advantages in the proposed approach are its flexibility of working with different types of machinery and the possibility to copy parts of the model (=sub-models) from one industrial plant to another where similar components are used.

References:

[1] Rao B.K.N., Handbook of Condition Monitoring, Elsevier Science Ltd., Oxford, 1996, ISBN 1-85617-234-1

[2] Schneiderman B., Designing the User Interface, Strategies for Effective Human-Computer Interaction, Addison-Wesley Longman Inc., 1998, ISBN 0-201-69497-2

[3] DeFatta J., Lucas J.G. and Hodgkiss W.S., Digital Signal Processing, A System Desing Approach, John Wiley & Sons, Inc., New York, 1988, ISBN 0-471-63765-3

[4] Jantunen E. and Jokinen H., Reduction of Data Needed in an Expert System for Condition Monitoring of FMS Using Regression Analysis Techniques, Proceedings of Comadem '96 1996. University of Sheffield, Sheffield

[5] Milton J.S. and Arnold J.C., Probability and Statistic in the Engineering and Computing Sciences, McGraw-Hill, 1986, New York, ISBN 0-07-042351-2

[6] Rao V.B. and Rao H.V., C++ Neural Networks and Fuzzy Logic, Management Information Source Inc., New York, 1993, ISBN 1-55828-298-x

[7] Schalkoff R., Pattern Recognition, Statistical, Structural and Neural Approaches, John Wiley & Sons, Inc., 1992, ISBN 0-471-52974-5

[8] Lumme V.E., Practical Neural Net Applications in Condition Monitoring, Proceedings of Comadem '97, Vol. 2 1997. Espoo, Finland, ISBN 951-38-4563-X, pp. 162- 171

[9] Rahmel D., Building Web Database Applications with Visual Studio 6, Osborne/McGraw-Hill, Berkeley, 2000, ISBN 0-07-212094-0