NINETEENTH EUROPEAN ROTORCRAFT FORUM

Paper No. L2

FLIGHT STATE RECOGNITION WITH NEURAL NETWORKS-A STEP TO AN OVERALL USAGE MONITORING SYSTEM

by

E. BRAND, B. KRUMME
EUROCOPTER DEUTSCHLAND GMBH
MÜNCHEN, GERMANY

September 14-16, 1993
CERNOBBIO (Como)
ITALY

ASSOCIAZIONE INDUSTRIE AEROSPAZIALI
ASSOCIAZIONE ITALIANA DI AERONAUTICA ED ASTRONAUTICA

FLIGHT SATE RECOGNITION WITH NEURAL NETWORKS - A STEP TO AN OVERALL USAGE MONITORING SYSTEM

by
E. BRAND and B. KRUMME
EUROCOPTER DEUTSCHLAND GMBH
MÜNCHEN , GERMANY

Abstract

Global practice in helicopter life time calculation is to use measured flight test data and a predetermined mission profile which is set up once for each helicopter type to cover all the different operator missions.

In reality, the operators fly unknown profiles and the design profile has to be chosen very conservatively. This means that manoeuvres have to be included which are more severe than those normally flown by the operator. To establish more realistic mission profiles, neural networks were applied to do flight state recognition in the operator's helicopter. A HUMS is sampling and calculating the data and delivers it to a ground station which enables the operator to calculate individual life times for each helicopter and each component in service.

Several selected input values (usually used values in common helicopter) have been investigated to get the correlation to the flight profile. In addition, different types of neural networks were considered.

The paper presents first results showing that the neural network was able to identify steady flight states without any problem and gives an outlook on further investigations which are necessary for more reliable recognition of transient flight conditions (manoeuvres) and also for the record of weight and centre of gravity influencing the structural loads.

1. Introduction

Today all components and systems of a helicopter have fixed life times and fixed TBO's (Time Between Overhaul). The time is counted according to the flight-logbook in which the pilot is writing the take-off and landing times. This times are summed up and compared with the fixed life time of the components.

The life times are calculated by the manufacturer with flight load data from the certification test flights and with a mission profile (distribution of flight states) which was set up once to cover all different operators (Fig. 1). The result is a list of components with fixed life times and TBO's.

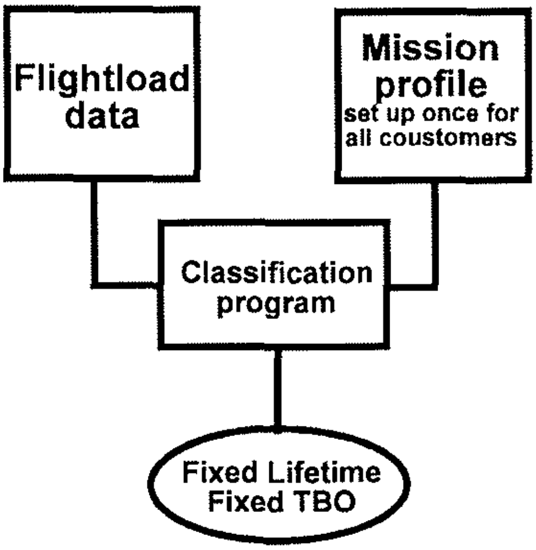

Fig. 1: Common method of life time calculation

2. HUMS General

A HUMS (Health and Usage Monitoring System) can bring a benefit to an operator. The distribution of the Direct Operating Costs (DOC) show that there is a large area in which an HUMS or a UMS (Usage Monitoring System) will reduce costs and save money. Maintenance which can be done later, saves money. A component which has not to be bought , saves money. So about 50% of the DOC can positively be influenced by a HUMS (Fig. 2)

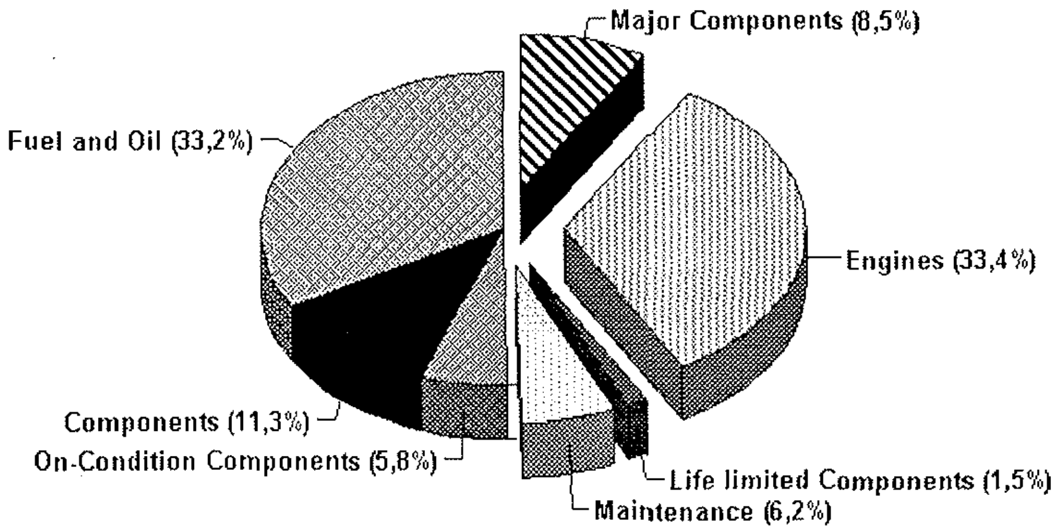

Fig. 2: DOC of a Helicopter and the parts of the DOC which can be influenced by HUMS (BK117 statistical data 1992)

When the mission is composed of a high percentage of manoeuvres, the summation of flight loads reaches the designed load limit earlier than calculated. So, a HUMS which can recognise this, improves safety. When the helicopter is flown very smoothly the life time limit can be increased and this is a benefit for the operator (Fig. 3).

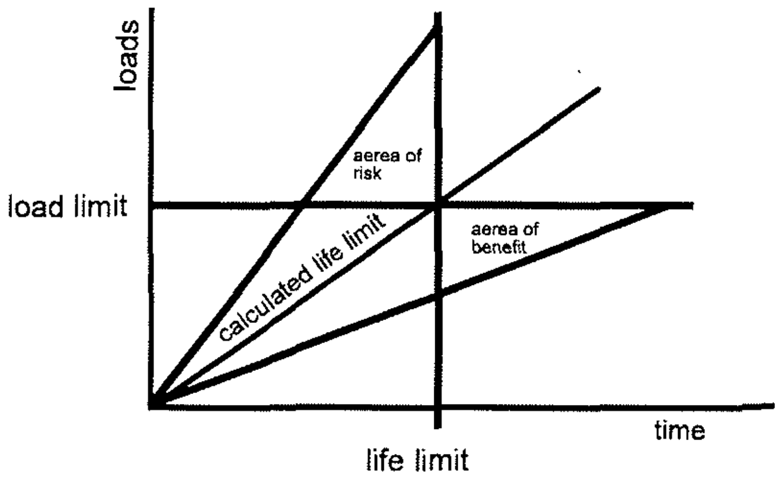

Fig. 3: Risk and benefit of a calculated life limit

3. USAGE Monitoring

3.1 Flight State Recognition

As shown in chapter 2 the life time calculation consists of two main parts:The flight load data and the mission profile. The flight loads can not be influenced. Each flight load results from a specific flight state. The distribution of the flight states and the mission profile influences the load collective. Recognising the actual flight states and summarising them to a mission profile allows to calculate an individual life time (Fig. 4). The question is how to recognise the actual flight state. One method is to use a neural network for this task.

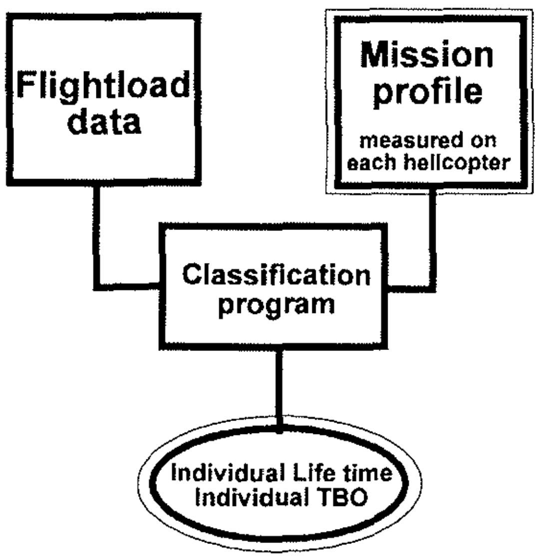

Fig. 4: Life time calculation with flight state recognition

3.2 Using Neural Network for Flight State Recognition

3.2.1 How does a Neural Network work

The Neural Network software simulates the simplified function of human brain with neurones and synapses (Fig.5). The input layer reads the input signals like flight velocity, control angles etc. Each neurone of the input layer is connected to each neurone of the next layer, the hidden layer. Each connection is done by a simple mathematic calculation. At the end the output layer generates a code which represents a specific flight state.

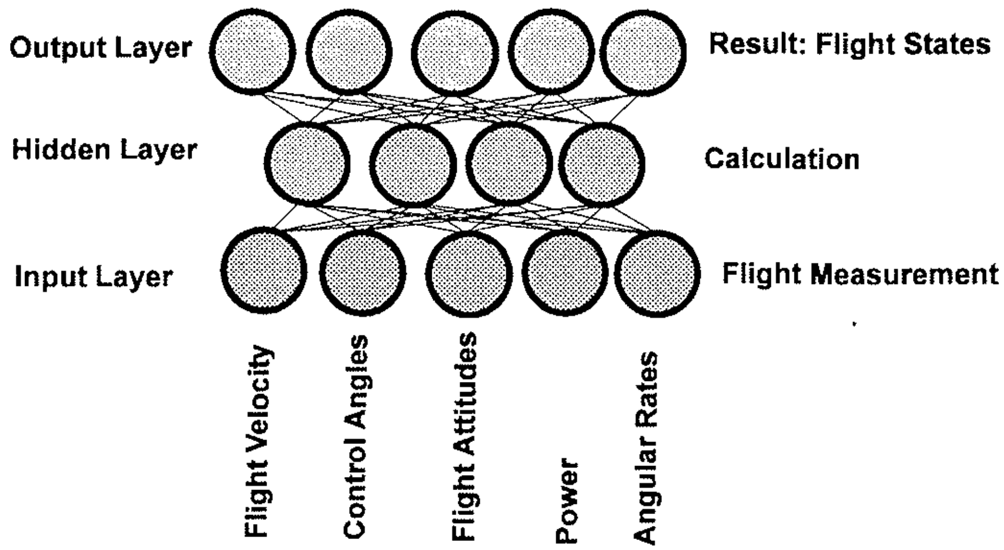

Fig. 5: The structure of a neural network with neurones and connections between each neurone

To provide the association between the input data and the flight state, these data are presented to the in- and output layers of the neural network. In the so called learning phase the neural network changes the values of the connections between the neurones. In this way the neural network adapts itself until it is able to associate input values and flight states in the right manner. Due to the mathematical model, the neural network used, is called back propagation network. After the learning phase, the neural network is tested with different data.

3.2.2 Flight State Recognition with Neural Networks

Sensors have to be installed in the helicopter, or as far as possible already installed sensors can be used. Amplifiers lead the signals to the input layer. An example of a signal combination is shown in Fig.6 . The left hand side is an example for a steady flight state, the right hand side for a transient flight. These examples demonstrate that a transient flight is more difficult to recognise than a steady state. The signals change very quickly as time proceeds and the neural network needs a longer time section to recognise the signals in the right way.

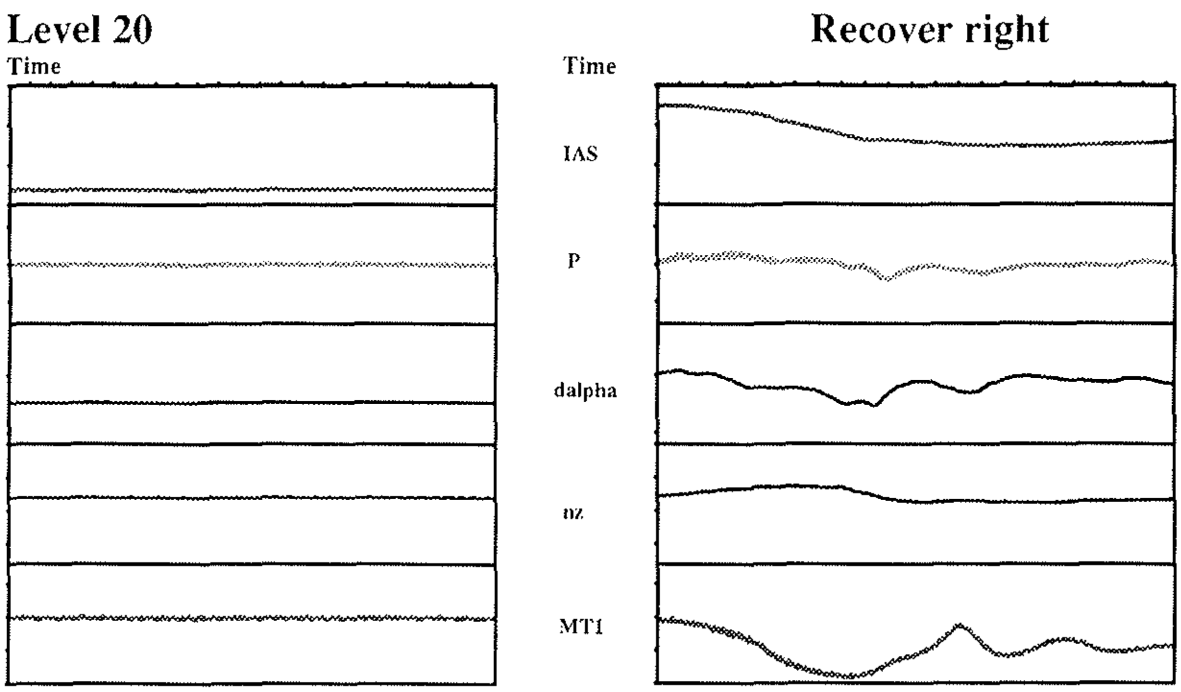

Fig.6: Example of measurement input signals for steady and transient flight states

The neural network can be implemented together with amplifiers and a memory in a "black box". Additionally the duration time of each flight state is stored (Fig.7). The stored data will be transferred to a ground station. There individual lifetime, TBO, historical files and trend analysis are calculated to support the maintenance services.

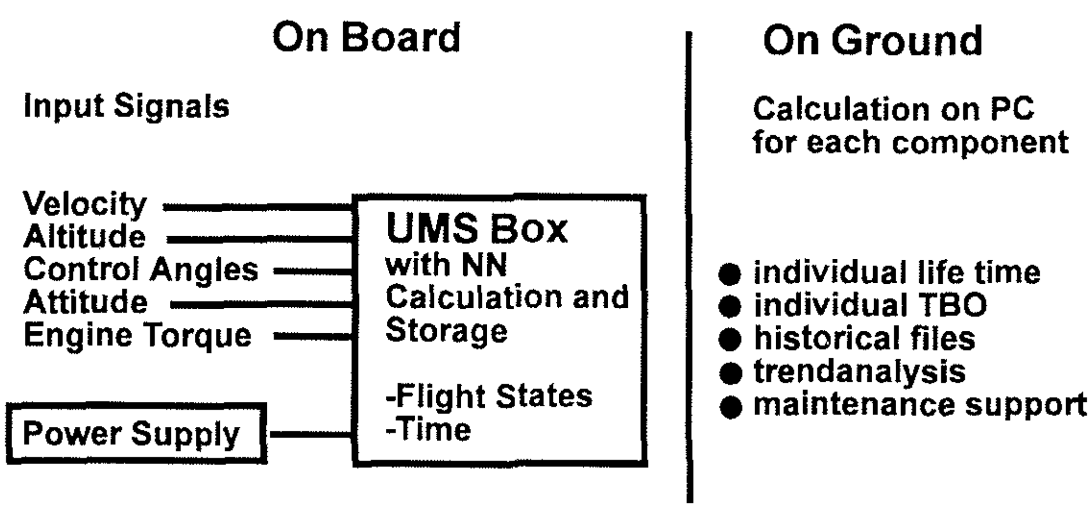

Fig. 7 Principle structure of a usage monitoring system for flight state recognition

3.2.3 Test Results

The neural network has been tested in a laboratory test with real flight test measurement data. The following tables show two recognition matrixes (Fig.8) for steady state flights and for transient flights. The test data which were used are different from the learning data.

Recognition matrix for steady flight states

Recognised Flight States		1	2	3	4	5	6	7	8	9	10	11	12	13
Descent	13													100
Turn right 45°	12												100	
Turn left 45°	11											100		
Turn right 30°	10										100			
Turn left 30°	9									100				
Turn right 15°	8								100					
Turn left 15°	7							100						
Level 100 kts	6						100							
Level 80 kts	5					100								
Level 20 kts	4				100									
Climb	3			100										
HOGE	2		100											
HIGE	1	100												

Input Flight States

Fig. 8a: Test Results for Steady Flight States

Recognition matrix for transient flight states

Recognised Flight States		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Turn left-right	16															13	82
Turn right-left	15															77	8
Stick inp.left	14														100		
Stick inp.ri	13													100		3	
Stick inp.fwd	12												0				
Stick inp.aft	11											0					
Pedal inp.left	10										100	8					
Pedal inp.ri	9							2		100							
Recovering	8							4	100			16					
Recover left	7				70			93									
Recover ri	6			90			100										
Pull up	5				30	100						29					
Pull up left	4				0												
Pull up ri	3			10													
Quick stop	2		87														
Push over	1	100															

Input Flight States

The missing flight states in this matrix are associated to steady flight states

Fig. 8b: **Test Results for Transient Flight States**

On the bottom numbers for the flight states are shown which were given to the neural network. On the left hand side the same numbers are indicated by the name of the flight state. If the flight state is recognised correctly 100% is written in the corresponding box. The recognition matrix for the steady flight states shows that all flight states are recognised by 100%.

For transient flights, the situation is more difficult. The numbers which are not placed in the diagonal illustrate that some manoeuvres are very alike due to the combination of the input signals. Consequently, they produce similar loads. Regarding the load aspect the number of flight states can probably be reduced. This will be a next step in our development.

3.2.4 System Application

Concerning hardware the system can be realised very easily. "Black boxes", which are capable of measurement amplifying, calculation and storage already exist on the market. Only the software has to be implemented (Fig.9). Data can be transported with a memory card to the ground station. There, a program has to calculate all the necessary data to support the operator in his work (Fig. 10).

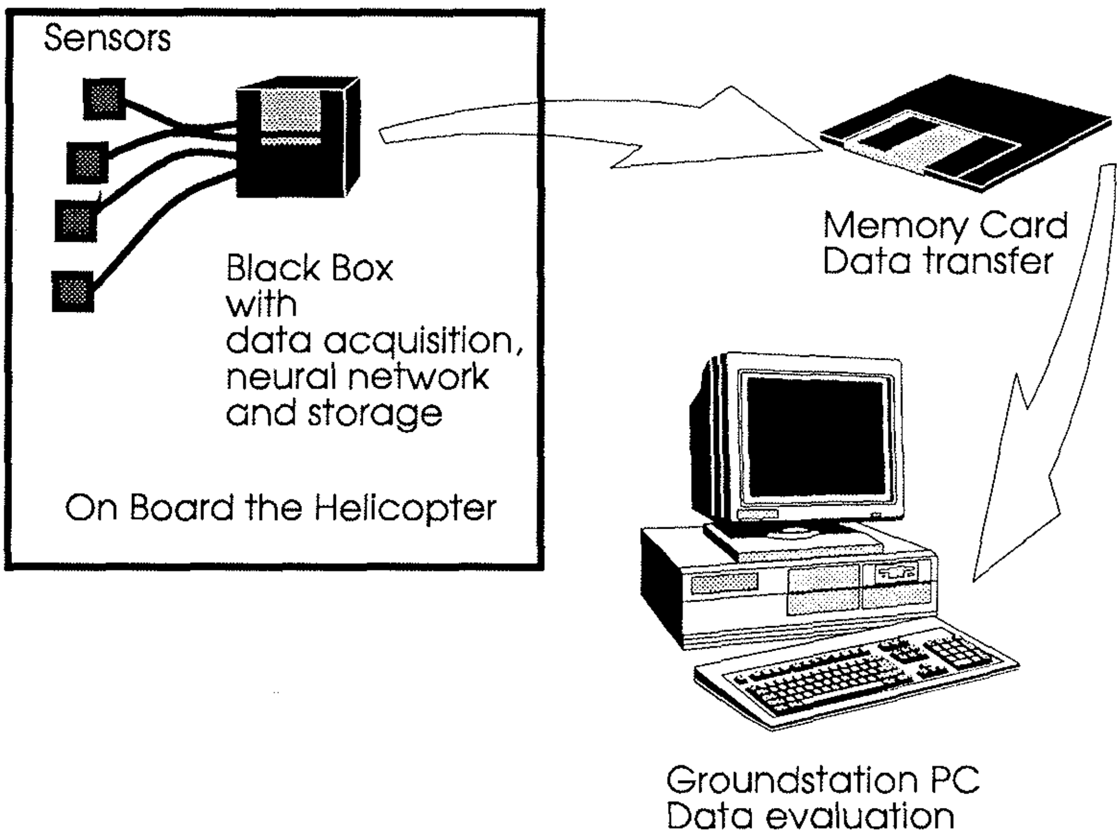

Fig. 9: Example for a system application including the ground station

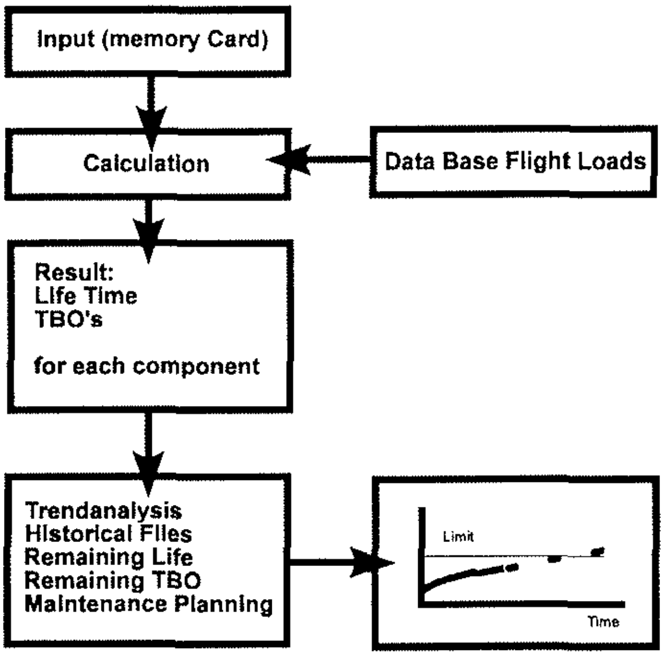

Fig. 10: Functional structure of the ground station software

4. Safety Aspects

The question of safety arises with all HUM systems. What is to do if the system fails? Of course can be installed double sensors, double processors, double memories and two independent software's, but this will be too expensive for such a system. The flight state recognition is not a flight critical system, so the flight task has not to be interrupted when the UMS fails. The system can recognise e.g. the sensor which has failed and the duration of the failure. Afterwards in the ground station the gap can be filled with help of the trend analysis.

5. Conclusion

Neural networks have been used for flight state recognition. In a first period, it was possible to show that not only steady flight states can be recognised but also transient flights.

Now the network has to be trained to recognise all flight states which are necessary for the load classification.

After that, a test period will start with on line data from flight test measurement.

Finally the network will be implemented into a "black box" and the ground station system has to be set up.

References

1. R. Bauman, F. Gasper, M. Spengler, Neuronale Netze LOG IN 12 Heft 1, 1992

2. H. Ritter, T. Martinez, K. Schulten, Neuronale Netze, Addison Wesley Publishing Company, Bonn, München, Reading Mass., 1991

3. J. Lawrence, Neuronale Netze, Computersimulation Biologischer Intelligenz, Systhema Verlag GmbH, München, 1992