ARTIFICIAL NEURAL NETWORKS FOR FAULT TOLERANT SENSOR SYSTEMS

D. Martens
Institute of Flight Guidance and Control
Braunschweig Technical University
Hans-Sommer-Str. 66
D-38106 Braunschweig, Germany

19th ICAS Congress/AIAA Aircraft Systems Conference, Anaheim, USA
September 18–23, 1994

Abstract

Artificial Neural Networks (ANNs) are a new approach for parallel information processing. They may be assumed as black boxes that admit inputs and produce outputs. The idea is based on the structure and function of a biological brain, which is able to recognize and classify objects, feelings or smells as well as it is able to learn, coordinate and control movements among performing other very complex tasks. Today's classical computers usually consist of one or a few powerful processors that follow all instructions sequentially. For every single problem a special piece of software is needed. In contrast to this a biological brain has a massively parallel structure in which a large number of cellular units (neurons) process simultaneously the information acquired by the body's sensory organs. And all tasks are accomplished by the same general piece of software, which is based on adjustable weights and patterns between the neuronal interconnections.

At the Institute of Flight Guidance and Control of Braunschweig Technical University some feasibility studies as well as first simulations in the field of sensor monitoring and diagnosis with ANNs have been performed. The first promising results are presented in this paper.

Introduction

Due to the efficiency and high flexibilty of neural processing systems the idea arised to design so called artificial neural networks that resemble in a strong simplified way the neural architecture of a biological brain . Beside this flexibility it must be pointed out that the outputs are produced almost instantaneously in comparison to serial architecture processors, especially when the ANN is implemented as hardware (e.g. as a semicustom chip).

As ANNs are trainable dynamical systems, their use is particularly interesting in those fields where the functional relation between input and output is mathematically difficult to describe or not known at all. They are generally used to perform pattern recognition and classification tasks (e.g. image and voice recognition) as well as adaptive (fuzzy) control. Due to the parallel information processing ANNs are inherently fault tolerant being suitable for guidance and control tasks.

As an example, simulation results of redundant sensor systems – e.g. inertial sensors, GNSS[†]-satellites – under real operation conditions (i.e. noise, sensor failures, etc.) showed that ANNs may be considered to be a potential new technology for next generation guidance and control systems. When using redundant sensor systems not only the detection but also the localization of a failure is an important feature to increase reliability: by localizing a malfunctioning sensor it's signals can be ignored. For this task the use of ANNs seems to be very interesting as they are

[†]GNSS: Global Navigation Satellite System

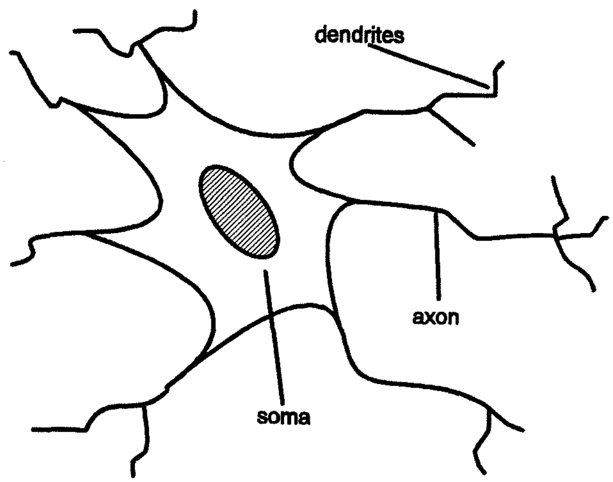

Figure 1: Scheme of a biological neuron

able to keep the time for detection and localization of a failure as well as for the following system reconfiguration very short.

Biological Background

The human brain consists of approximately 10^{10} neurons. The most important components of a neuron are the body (soma), the axons which end in hundreds of thousands of ramifications called dendrites (fig. 1). These dendrites are connected to other cells via so-called synaptic end heads. Today, it is known that the electrochemical, biochemical or morphological modification of these synaptical attachments produces a change in the data stream of the brain, which causes a change in behaviour. The electrochemical state of a synaptical connection is responsible for the intensity at which an electrical impulse is transmitted from one neuron to another. The effectiveness of a synaptical connection state depends on the activity of the transmitting neuron as well as on the activity of the receiving neuron. This yields a coupled, non-linear dynamical stream of information[1]. Today, it is not known exactly how this process works.

It is remarkable that today's digital computers have cycle times of a few nanoseconds, whereas the comparable cycle time of a neuron is of some milliseconds. Despite this huge difference in processing time neural networks are superior to digital computers in many applications. The reason for this is the massively parallel processing of information compared to the sequential processing of conventional computers.

Models of ANNs

In human thinking there are two basically different processes:

1. conscious thinking
2. unconscious intelligence

The first process is comparable to the function of digital computers where calculations are performed in sequential steps of a programm retained in the computer's memory. To solve a particular task with this technique an algorithm that takes into account all specific problems has to be developed à priori.

In contrast to this programming technique an ANN is able to 'learn' the connection between input and output by producing a projection of the mathematical function. The quality of the results is dependent on the past history. This means that the unconscious intelligence may

- process huge amounts of data,
- point out special characteristics,
- associate characteristics,
- perform data reduction and
- perform movement control.

General structure

The unconscious thinking lead to the development of ANNs consisting of simple interconnected processing elements (fig. 2). The processing elements are generally grouped in layers and they may be excited or inhibited by positive or negative weights respectively.

Mathematical model of a neuron

Fig. 3 shows the function of an artificial neuron. The output signals x_j of the n processing

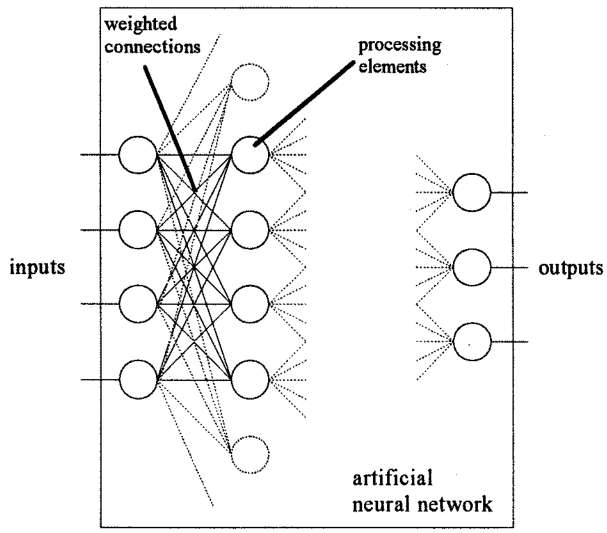

Figure 2: Basic structure of artificial neural networks

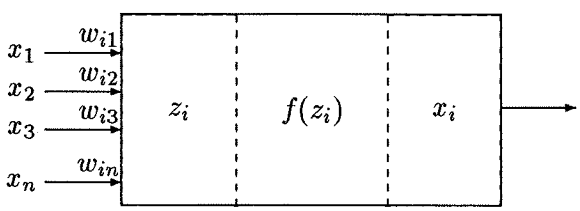

Figure 3: Basic function of a processing element

elements of the foregoing layer are the input of processing element i. These signals are multiplied by the weight factors w_{ij} and processed by the input function z_i. A transfer function $f(z_i)$ defines the activation state of the processing element which is decisive for the value of the output x_i. The most important input, transfer and output functions are summarized in table 1.

Application of ANNs

ANNs are succesfully applicable especially to such problems where a knowledge of mathematical rules or parameters is not available. In such cases it is difficult to develop an algorithm for conventional digital computers. Due to their ability to learn it is possible to apply ANNs on problems where undetermined or contradictory input data is available[2][3]. Non-linear processes may be reproduced as well. Thus, there is a great variety of possible applications[5][6]. Some examples are stated below:

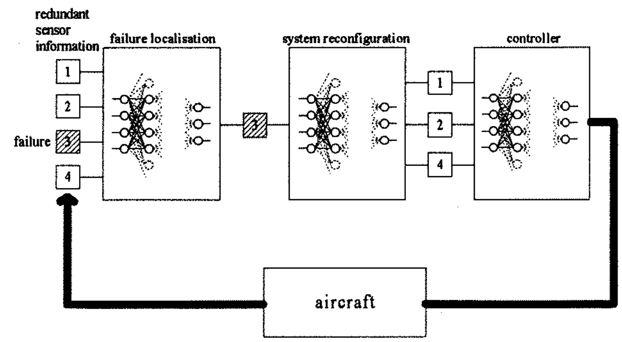

Figure 4: Fault tolerant flight management system

- medical & neurological research
- communication technology (filtering, data compression, speech recognition, etc.)
- assurances and banking (signature identification, stock market prediction, etc.)
- quality control
- system monitoring
- integration of sensor data
- guidance and control
  - adaptive flight control
  - target recognition
  - decision making
- optimization techniques

Table 2 summarises the most important advantages and disadvantages of ANNs[4].

At the Institute of Flight Guidance and Control the ANN technology was applied succesfully for monitoring two different kinds of sensor systems:

1. monitoring of inertial sensor data
2. monitoring of GNSS data

Such kind of monitoring modules are a necessary element of any modern fault tolerant flight management system (fig. 4).

The reliability of any sensor system depends on the ability to recognize failures and to reconfigure the system in such a way that the failure is

Function	Name	Mathematical expression
Input	Sum	$z_i = \sum_{j=1}^{n} w_{ij} x_j$
	Product	$z_i = \prod_{j=1}^{n} w_{ij} x_j$
	Maximum	$z_i = \max(w_{ij} \cdot x_j)$
	Minimum	$z_i = \min(w_{ij} \cdot x_j)$
	Majority	$z_i = \sum_{j=1}^{n} S_j \quad S_j = +1 \; \forall \; w_{ij} \cdot x_j > 0$ $S_j = -1 \; \forall \; w_{ij} \cdot x_j \leq 0$
	Sigma-Pi	$z_i = \sum_{j=1}^{n} w_{ij} \cdot \prod_{k=1}^{m} x_{jk}$
Transfer	Linear	$f(z_i) = z_i$
	Sigmoid	$f(z_i) = \frac{1}{1+e^{-cz_i}}$
	Tangent hyperbolicus	$f(z_i) = \tanh(c \cdot z_i)$
	Sine	$f(z_i) = \sin(c \cdot z_i)$
	Signum	$f(z_i) = \mathrm{sgn}(z_i)$
	Step	$f(z_i) = 1 \; \forall \; z_i > 0$ $f(z_i) = 0 \; \forall \; z_i \leq 0$
	Linear Threshold	$f(z_i) = z_i \; \forall \; z_i > 0$ $f(z_i) = 0 \; \forall \; z_i \leq 0$
Output	Direct	$x_i = f(z_i)$
	Winner Takes All	$x_i = h[f_k(z_i)] \; \forall \; f_k(z_i) = \max(f(z_i))$ otherwise $x_i = 0$; h: any function

Table 1: Functions for modelling processor elements[2]

Pros	Cons
1. no algorithms needed: time saving	1. not applicable to all problems (fuzzy data processing)
2. little software needed	2. à priori input and output data generally needed (supervised learning)
3. inherent parallelity enables fast solutions	3. time consuming search for usable network type
4. robust at noisy signals	
5. fault tolerant (graceful degradation)	

Table 2: Pros and cons of ANNs in comparison to conventional algorithms

eliminated. The recognition and reconfiguration time is a very important factor during this process in order to avoid unstabilizing phase lags in the control system.

Krogmann[4] conceived a system of ANNs able to recognize failures of inertial sensor systems. For this purpose a number of redundant sensors for the angular velocity (e.g. gyros) as well as for the linear acceleration (accelerometers) are used. The axes of the sensors are skewed in such a way that each sensor monitors several axes of a vehicle's motion. The task of the ANN system is to detect faults and performance degradations and to localize the defective sensor among the redundantly available ones.

Fig. 5 shows the arrangement of the ANN signal processing modules needed for this purpose. The measurement vector $\underline{m}$ comprises the sensor outputs and is a direct function of the vehicle's motion state. The output of this module is a vector called feature vector $\underline{v}$ containing a projection of the measurement vector $\underline{m}$. As this projection may be determined analytically for a specified fixed sensor geometry the weights of the first ANN module are hard wired. The second ANN module is in charge of the fault detection and localization by performing a classification of the feature vector $\underline{v}$. The classification vector $\underline{s}$ contains the information about which of the sensor axes produces faulty data. Correspondingly, only one component s_i of $\underline{s}$ has a high activation ($s_i \approx 1$) whereas the other components have a low activation ($s_j \approx 0,\ j = 1, ..., n,\ j \neq i$). In this case the functional relation between input $\underline{v}$ and output $\underline{s}$ is not known at all. The network is 'trained' to perform this projection by optimizing its interconnection weights with a series of input vectors and corresponding output vectors until the calculated output resembles the ideal vector $\underline{s}$.

Simulation results with this system are presented in fig. 6. It is clearly to see how the network recognizes a faulty sensor axis.

A similar system was developed at the Institute of Flight Guidance and Control to monitor the integrity of GNSS data. If the use of such navigation systems is desired for air navigation a fast localization of faulty satellite signals is im-

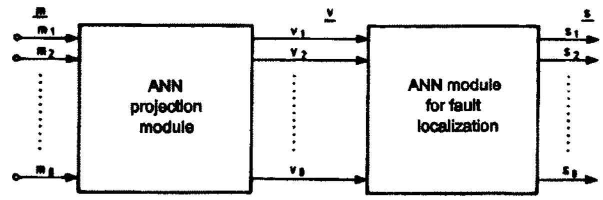

Figure 5: ANN system for sensor monitoring purposes

portant for operational aspects. The time within which such faults have to be identified depends on the flight phase (en-route, landing approach, etc.) and is relevant for certification aspects. For CAT III landings an identification time of less than two seconds is required. Today's GNSS (GPS and GLONASS) do not allow for such a fast integrity monitoring so that this has to be performed at the receiver with the available satellite data. As today's comprehensive statistical methods for integrity monitoring consume a lot of processing time on digital computers the use of ANNs for the monitoring of satellite data is investigated. First promising simulation results are depicted in fig. 7.

The first two diagramms on top of fig. 7 show the residuals of a ten channel GPS receiver. After 260 and 470 seconds of simulation time respectively a step error is added to the ranges of the receiver channels 3 and 5. Each of these errors spreads through the navigation solution into the residuals of all other channels as it is shown in the diagramms. These residuals, as well as some information about the satellites relative position, are fed to an ANN module to localize the faulty receiver channels. The two lower diagramms of fig. 7 show the output signals of this integrity monitoring ANN system. As long as there is no fault the output signals of the ten receiver channels remain at a low activation level ($s_i \approx 0$). As soon as an error appears the monitoring system recognizes the faulty channel by activating the corresponding output signal ($s_{3,5} \approx 1$).

Concluding Remarks

The foregoing analysis demonstrates that it is possible to perform very fast monitoring of two

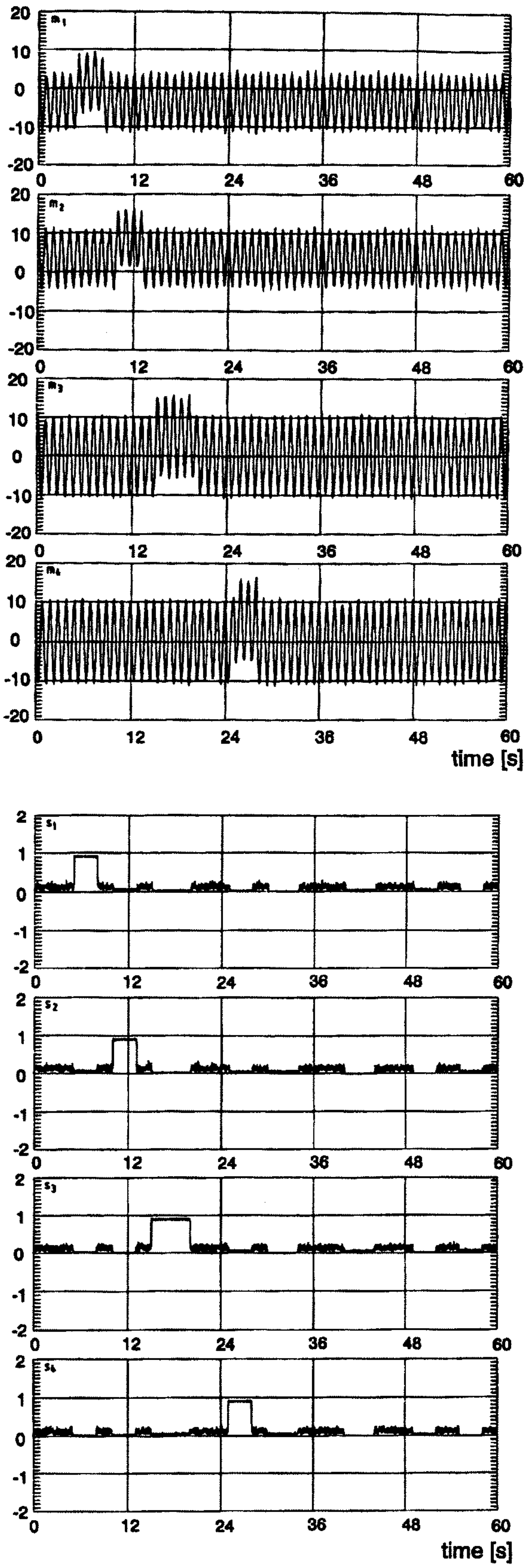

Figure 6: Localization of faulty gyro signals with ANNs (above: measurement signals m_1 to m_4 [°]; below: localization signals s_1 to s_4)

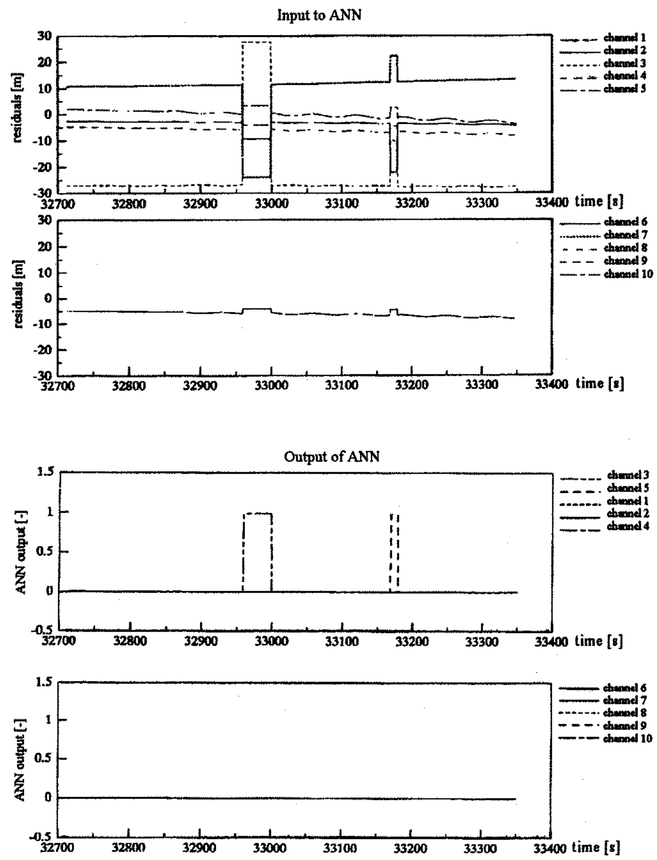

Figure 7: GNSS integrity monitoring with ANNs

special sensor systems with ANNs. Due to the broad application potential of ANNs other areas may make use of this fault tolerant data processing technique.

Nevertheless, there is still a lot of research to do in the area of neuro-informatics. This concerns mainly the stability and the rules for the search of usable network types to solve practical problems. The latter still is very often a time consuming try-and-error procedure. Once the solutions to these problems are found it will be possible to answer many questions about complex dynamical processes.

References

(1) H. Ritter, Th. Martinetz, K. Schulten: Neuronale Netze: eine Einführung in die Neuroinformatik selbstorganisierender Netzwerke – Addison Wesley, Bonn, München (u.a.), 1991

(2) E. Schöneburg, N. Hansen, A. Gawelczyk: Neuronale Netzwerke: Einführung, Überblick und Anwendungsmöglichkeiten – Markt & Technik Verlag, 1990

(3) B. Kosko: Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence – Prentice Hall, Englewood Cliffs, NJ 07632, 1992

(4) U. Krogmann: Introduction to Neural Computing and Categories of Neural Network Applications to Guidance, Navigation and Control – AGARD-LS-179, 1991

(5) J.L. McClelland, D.E. Rumelhart: Explorations in Parallel Distributed Processing: a handbook of models, programs and exercises – MIT Press, Cambridge, Mass., 1988

(6) T. Gutschow, R. Hecht-Nielsen: Advance Neural Network Architectures for Guidance and Control – AGARD-LS-179, 1991