

Performance Analysis and Assessment of Safety Critical Systems Using Ann

K. Sruthi, R. Krishnaprasad, Manju Nanda and J. Jayanthi

Abstract Many of the safety critical systems (SCS) require the failure analysis to ensure that the applications work properly and safely. Failure of such safety critical systems results in huge losses. Unmanned aerial vehicles (UAVs) are safety critical systems. System Safety Assessments (SSA) is employed for analyzing these systems. SSA monitors the degraded functionalities of systems. Sensor faults are the root causes for most of the system faults and need to be assessed properly. The availability of several tools for the design, implementation, and simulation of artificial neural networks (ANN) makes it easier to use in safety assessment applications. MATLAB provides ANN features to analyze the sensor faults. In this paper we propose ANN as a novel approach to ensure the system safety and analyze the failures due to the sensor faults. UAV groundspeed sensor is considered as a case study. MATLAB/Simulink tool suite is used to implement the sensor model. The model outputs are classified and the classified values of sensor are used for prediction. The predicted values are classified for verifying the future performance. Simulation results discuss the classified values, performance analysis, and predicted outputs of the sensor model.

Keywords UAV · ANN · SCS · SSA

K. Sruthi (✉)

CESP, Jawaharlal Collage of Engineering and Technology, Palakkad, Kerala, India
e-mail: 92sruthik@gmail.com

R. Krishnaprasad

CSIR-National Aerospace Laboratories, Bangalore, India
e-mail: krishnaprasad.rajana90@gmail.com

M. Nanda · J. Jayanthi

Aerospace and Electronics Division, CSIR-National
Aerospace Laboratories, Bangalore, India
e-mail: manjun@nal.res.in

J. Jayanthi

e-mail: jayanthi@nal.res.in

© Springer Nature Singapore Pte Ltd. 2018

N.R. Shetty et al. (eds.), *Emerging Research in Computing, Information, Communication and Applications*, https://doi.org/10.1007/978-981-10-4741-1_12

1 Introduction

Safety-critical systems are those systems whose failure could result in loss of life, significant property damage, or damage to the environment. Many modern information systems are becoming safety-critical in a general sense because financial loss and even loss of life can result from their failure [1]. Engineering of safety-critical systems is a complex task involving many technical fields. Unmanned aerial vehicles (UAVs), also known as drones, are aircraft either controlled by ‘pilots’ from the ground or increasingly, autonomously following a pre-programmed mission. It is one of most common example of safety critical systems. Due to their advantages, UAVs are used in many civil and military applications such as aerial surveillance, remote sensing, cargo delivery, etc. About 80% of flight incidents concerning unmanned aerial vehicles (UAV) are due to faults affecting propulsion, flight control surfaces, or sensors [2]. Here we are considering the UAV system as a set of sensors namely groundspeed sensor, angle of attack sensor, accelerometer sensor, angle of sideslip sensor, differential pressure sensor, etc.

Safety assessment is the process for assuring the safety and reliability of critical aeronautical systems. It uses probabilistic analysis to provide precise measures about the safety requirements of a system. The basic information used as input to these techniques are failure conditions and failure rates [3]. We define prognosis to be detecting the precursors of a failure, and predicting how much time remains before a likely failure. Algorithms that use the data-driven approach to prognosis learn models directly from the data, rather than using a hand-built model based on human expertise [4]. One useful approach is the data-driven approach, also known as the data mining approach or the machine learning approach, which uses historical data to automatically learn a model of system behavior [5]. So data-driven approach along with the safety assessment helps in the safe handling of UAV systems. Being the most famous data-driven approach, ANN finds its application in this case too. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process [6]. The sensor model values are given to classifier and predictor which is used for prognosis. Groundspeed sensor is implemented using MATLAB/SIMULINK model and the outputs are given to ANN algorithm. Groundspeed is the horizontal speed of the aircraft relative to ground. Groundspeed sensor gives reasonable results in calm weather conditions. It measures actual airspeed resulting in better throttle control and aircraft performance especially in windy conditions [7]. The classifier output specifies whether the sensor works satisfactorily or not. Predictor predicts the future output values.

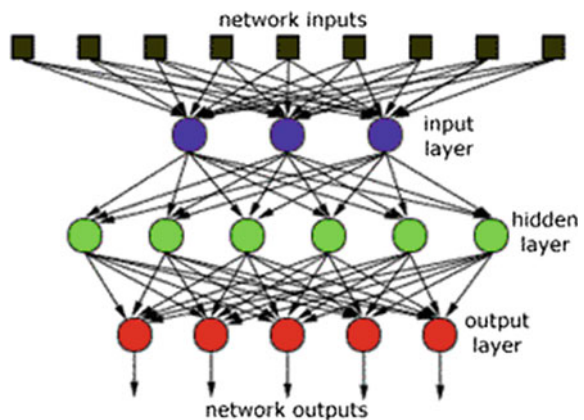
The paper organized in various sections as: Sect. 2 gives details about the ANN and tools, Sect. 3 explains block diagram and flowchart of the proposed approach, sensor model and algorithm details are given in Sect. 4 and results and conclusion are given in Sects. 5 and 6, respectively.

2 Related Work

Large variety of methods have been proposed for designing Safety analyzing systems. Some of the approaches are: pattern recognition, parameter estimation [8], observers (Luenberger observers, extended Luenberger observers, and unscented observers) [9], filters (Kalman filters, extended Kalman filters, unscented Kalman filters) [10], multiple model adaptive estimation and intelligent technology [11]. Intelligent technology consists of fault diagnosis such as expert system method, graphical method, soft computing, and distributed artificial intelligence. Graphical systems includes fault tree analysis and Bayesian network. Soft computing method consists of rough set, artificial neural network, support vector machine, fuzzy logic [12], artificial immune system, and decision tree are included. Among all these we are using ANN method. Bayesian networks are used to describe the relationship among project state impact factors, decisions to be taken and the satisfaction levels related to each decision. Approximated results are available in an efficient way while using Neural networks. An Artificial Neural Network (ANN) [13] is an information processing paradigm that is stimulated by the way biological nervous systems, such as the brain, process information. Usually, neural networks operate in two phases. The first phase is a learning phase where each of the nodes and links adjust their strength in order to match with the desired output. A learning algorithm is in charge of this process. When the learning phase is complete, the neural network (NN) is ready to recognize the incoming information and to work as a pattern recognition system (Fig. 1).

NN-based schemes benefit from factors such as; online learning algorithms, ability to model nonlinear systems and are programmable in a few lines of code [14]. There are several open source tools are available for ANN like open NN, just NN, Neural Machine, plug & Score, FANN, etc. Some of them are useful for commercial purpose while some others help in research works [15]. MATLAB is having lot of functionalities and tutorials are also available. So MATLAB is used

Fig. 1 ANN general architecture



for implementing the proposed method. The nonlinear autoregressive network with exogenous inputs (NARX) is a recurrent dynamic network, with feedback connections enclosing several layers of the network. The NARX model is based on the linear ARX model, and commonly used in time-series modeling.

3 Proposed Approach

The block diagram and the flowchart is explained in this section.

3.1 Block Diagram

Here the inputs are given to MATLAB SIMULINK model. These models are built according to the characteristic equation of each sensor. The outputs of sensor models are given to the neural network where the classification and prediction takes place. The predicted values are again given to the classifier which helps in evaluating efficiency of the system. The input values are decided on the basis of range of each individual values of the sensors (Fig. 2).

The proposed approach is relatively simple and precise than other safety assessment techniques. It checks the status of the system and along with that it monitors the future performance also. So, this approach helps in proper working as well as the long time life period of the system (Fig. 3).

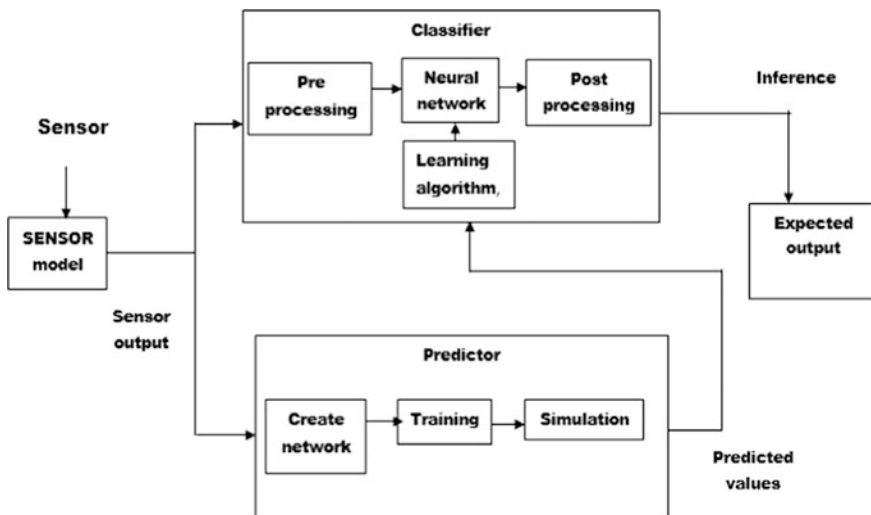


Fig. 2 Block diagram of the proposed approach

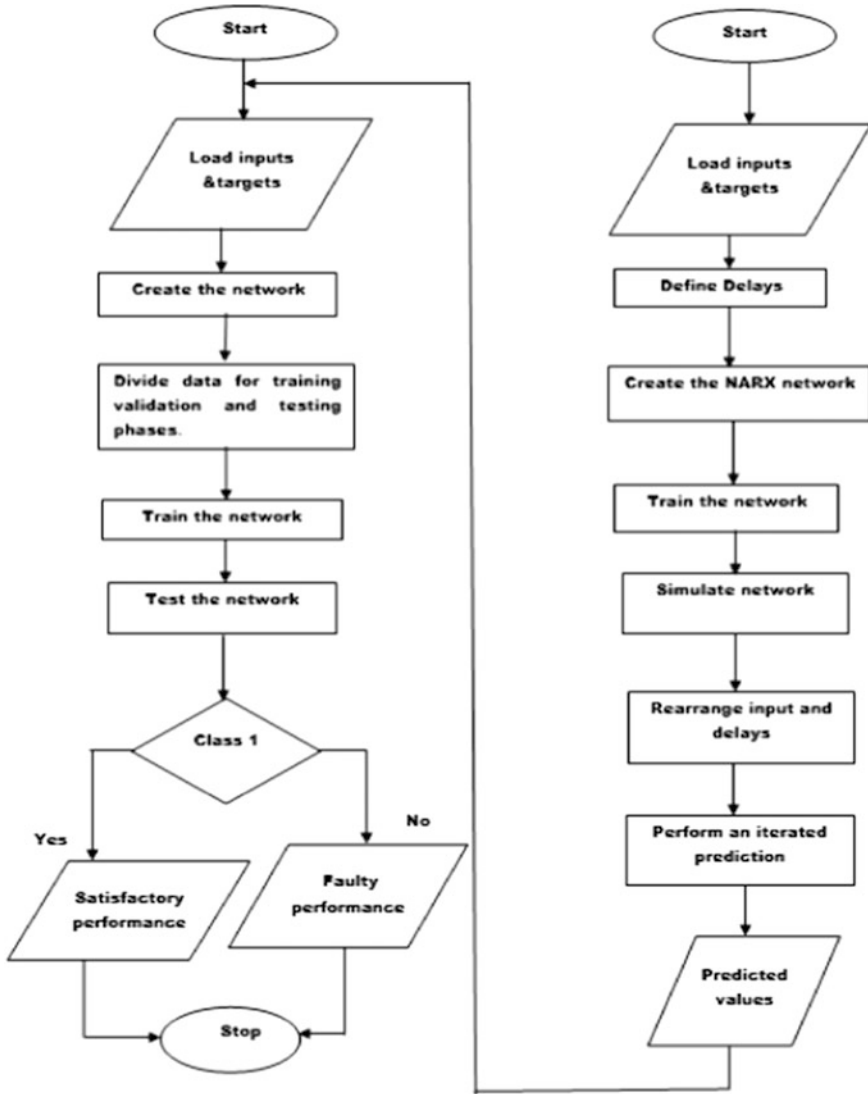


Fig. 3 Flowchart representation of system

3.2 Flowchart

The flowchart of the implemented program is given below. In the flowchart the general workflow is explained. In the case of classifier, after loading inputs and targets, networks is created, trained, and tested with real time data. The data is categorized based on the range of output values. If it belongs to class 1 the system

gives satisfactory performance otherwise it indicates some failure condition. In the predictor part, the NARX network is created, trained, simulated, and used for iterated prediction. The predicted values are again given to classifier network to check the working in the future slots.

4 Implementation

In this section classification and prediction algorithms are explained briefly along with the explanation of sensor model.

4.1 Modeling of Sensor

The sensor model has been created based on the characteristic equation. We have implemented the model for groundspeed sensor and its out values are processed accordingly. All the other sensors mentioned in Sect. 1 can be implemented similarly. Consider the case of ground speed sensor. The characteristic equation is given below:

$$Vg = \sqrt{(v^2 + [Vw]^2 + 2.v.Vw\cos(\Psi - Xw))} \quad (1)$$

where, Vg = groundspeed, Vw = wind speed, v = airspeed, Xw = wind direction and Ψ = yaw value.

The database is created according to the equation and range of value of each input. The inputs are imported to the Simulink model. The simulated outputs are saved in the work space. The outputs of the Simulink model are given to the ANN algorithm (Fig. 4).

4.2 Classification Algorithm

In this case the inputs are the groundspeed monitored from the output of the Simulink model. The targets are the classes which indicate to which category the input given belongs to. Pattern recognition networks are feedforward networks that can be trained to classify inputs according to target classes. The target data for pattern recognition networks consists of vectors of all zero values except for a 1 in element i , where i is the class they are to represent. Processing functions transform user input data to a form that is easier or more efficient for a network.

Training phase: The first step in training is to divide data for training, validation, and testing. Here we are using the method of sampling for this. The

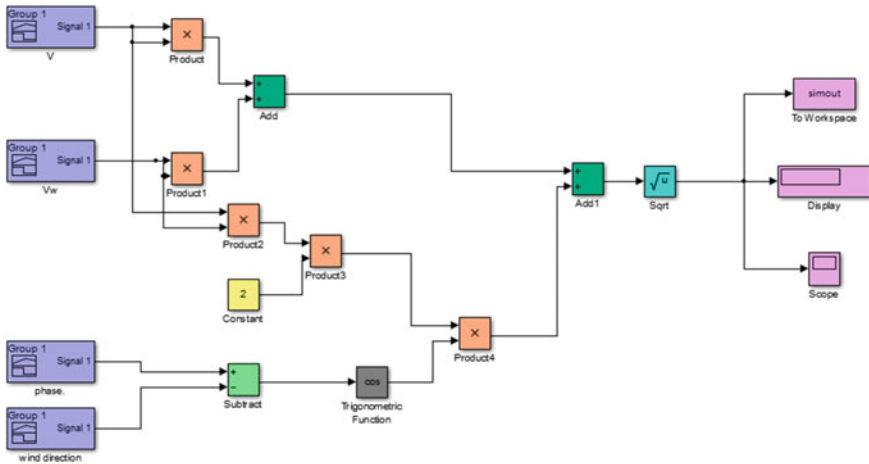


Fig. 4 Groundspeed sensor model

performance function used here is the mean square error mse—the average squared error between the networks outputs and the target outputs t . There are two different ways in which training can be implemented: incremental mode and batch mode. In incremental mode, the gradient is computed and the weights are updated after each input is applied to the network. In batch mode, all the inputs in the training set are applied to the network before the weights are updated. Here it is used the batch mode training with the train command which is faster and produces less errors. When training large networks and when training pattern recognition networks, ‘trainscg’ and ‘trainrp’ algorithms are good choices. Their memory requirements are relatively small, and yet they are much faster than standard gradient descent algorithms. Here ‘trainscg’ algorithm is used.

Perceptron concept: Perceptrons are simple single-layer binary classifiers, which divide the input space with a linear decision boundary. Perceptrons can learn to solve a narrow class of classification problems. Classification with pattern recognition gives more precise and accurate results.

4.3 Prediction Algorithm

When delay lines are used with multilayer feedback networks it can be used for prediction purposes. Some of the methods here used include NARX, NAR, etc.

Simple NARX method: Here after importing the inputs and targets, the delay values are specified. The typical workflow is to fully create the network in open loop, and only when it has been trained it is transformed to closed loop for multistep-ahead prediction. Thenby using ‘narx’ command the NARX network is created according to the hidden layer size given. Then the network is trained and

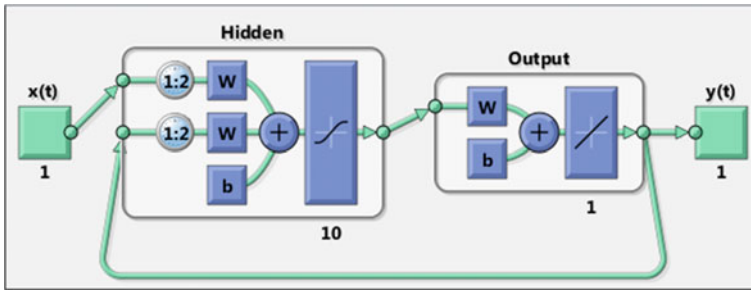


Fig. 5 Closed loop network for multistep-ahead prediction

simulated. The errors are plotted. For changing the series parallel configured network to parallel configuration ‘closeloop’ command is used. It helps for the multistep-ahead prediction (Fig. 5).

NARX network with training algorithm: Here the same NARX network is used along with the training steps which is explained in the classification part. The ‘trainlm’ algorithm is used to train the network. It is one of the fastest back propagation algorithm.

Function fitting networks: Function fitting is the process of training a neural network on a set of inputs in order to produce an associated set of target outputs. Once the neural network has fit the data, it forms a generalization of the input-output relationship and can be used to generate outputs for inputs it was not trained on. It can be used in the case of sensor values also.

NAR, Nonlinear input-output methods can also be used for the purpose. Only inputs differ in that case.

5 Results

The following images illustrate k-Means clustering applied to a set of randomly generated 2-d data points. The points are generated using a normal distribution centered at a mean location and with a constant standard deviation. The section contains the output samples as well as the performance graphs of classification and prediction. Classifier outputs are saved in workspace. The 1 in first column indicates class 1, i.e., satisfactory performance. Here instead of exact value, approximated value is an output like 0.99 (Fig. 6).

Performance graph: The graph shows the mse (mean squared error) of training, validation, and testing phases according to epochs. Performance is measured in terms of mean squared error, and shown in log scale. It rapidly decreased as the network was trained. Network gives the best performance at epoch 45.

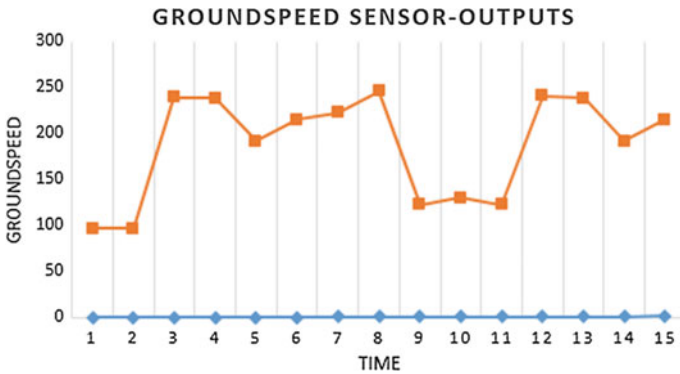


Fig. 6 Output waveform of groundspeed sensor

Error histogram: It shows an approximation of errors occurred at training, validation, and testing phases. Errors are calculated by subtracting outputs from targets.

Confusion matrix: The confusion matrix shows the percentages of correct and incorrect classifications. Correct classifications are the green squares on the matrices diagonal. Incorrect classifications form the red squares. If the network has learned to classify properly, the percentages in the red squares shows very small values, indicating few misclassifications. These performance measurements are available from the command window. These three plots, i.e., Performance plot, error histogram and confusion matrix are the outputs of classifier.

Also, Prediction can be done using the sensor values. The following graph shows the difference between the actual and predicted output. The figure illustrates the iterated prediction. The blue line is the actual output value of the sensor, and the red line is the value predicted by the NARX neural network.

The table shows the error values found in several prediction methods used. NARX with training gives better results (Table 1) (Figs. 7 and 8).

By analyzing these plots we can arrive at a conclusion about the working of sensor. If it is not working properly we can do failure preventive measures so that it helps to increase the lifetime of the system. The process is comparatively simpler while considering the Bayesian networks. Thus the classifier as well as predictor outputs together help for the safety assessment of system.

Table 1 Errors in each prediction methods

Prediction method	Errors in each time steps					
NARX with training	7.4411	6.8570	9.7674	7.9233	8.016	9.3242
Simple NARX	8.4411	9.8570	10.7674	8.9233	10.016	11.3242
NAR	15.441	19.8570	18.7674	12.923	13.016	15.3242
Nonlinear i/p-o/p	20.441	22.857	70.7674	40.9233	30.0167	27.3242

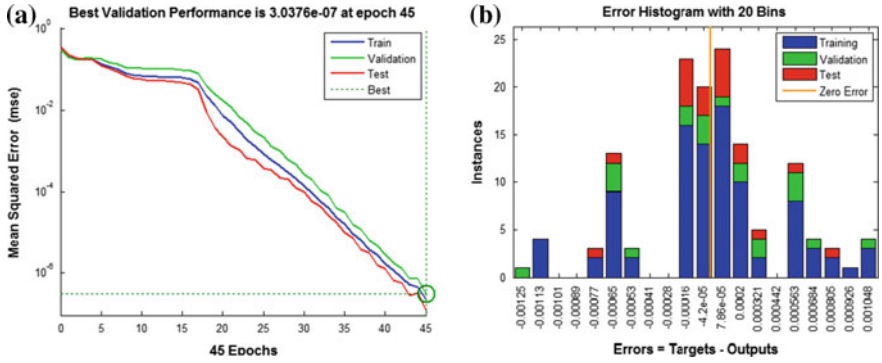


Fig. 7 a Performance graph, b Error histogram

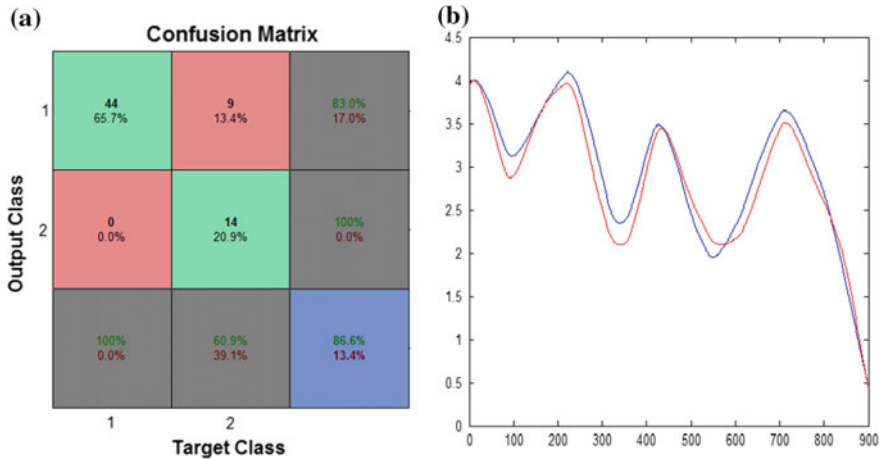


Fig. 8 a Confusion matrix, b Prediction plot

6 Conclusion

This paper explains the need of machine learning algorithms to tackle very large amount of data. Safety-critical systems are those systems whose failure could result in loss of life, significant property damage, or damage to the environment. From a software perspective, developing safety critical systems is required in significant areas such as specification, architecture, verification, and process. Due to the advantages, UAVs are used in many civil and military applications such as aerial surveillance, remote sensing, and cargo delivery. Being one of the most important example of safety critical system the safe handling of such UAV systems are very important. Safety assessment of such systems can result in efficient monitoring of their performance. Such safety systems help to improve the lifetime of UAV

systems. Analyzing the safety is not just concerned about the sensor faults. Many areas under which safety analysis can be used is under research. Use of ANN networks are very helpful while detecting the faults. However, the methods which use other some other techniques like Bayesian networks, Hidden markov models, etc., are also under the research.

References

1. Inseok Hwang, Sungwan Kim, Senior, Youdan Kim, and ChzeEngSeah. "A Survey of Fault Detetion, Isolation, and Reconfiguration Methods, IEEE transactions on control systems technology", Vol. 18, pp. 636–653, (2010)
2. Diogo Leite Rebouças, Fábio Meneghetti Ugulino de Araújo, André Laurindo Maitelli "Use of artificial neural networks to fault detection and diagnosis", ABCM Symposium Series in Mechatronics—Vol. 5, pp. 666–675, 2012
3. Lennon R. Cork¹, Rodney Walker¹, Shane Dunn "Fault Detection, Identification and Accommodation Techniques for Unmanned Airborne Vehicle".
4. Adriano Gomes, Alexandre Mota, Augusto Sampaio, Felipe Ferri, and Julio Buzzi "Systematic Model-Based Safety Assessment via Probabilistic Model Checking"
5. Iz Al-Dein Al-Zyoud and K. Khorasani "Neural Network-based Actuator Fault Diagnosis for Attitude Control Subsystem of an Unmanned Space Vehicle"
6. Iman Sadeghzadeh¹, and Youmin Zhang: "A Review on Fault-Tolerant Control for unmanned Aerial vehicle, St. Louis, Missouri", 2011
7. Bhaskar Prasad Rimal, Hyoim Shin and Eunmi Choi "Simulation of Nonlinear Identification and Control of Unmanned Aerial Vehicle: An Artificial Neural Network Approach", 2009, pp. 442–447
8. J Marzat, H Piet-Lahanier, F Damongeot, and E Walter, "Model-based fault diagnosis for aerospace systems", a survey, Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2011, pp. 1–31.
9. Iman Sadeghzadeh, and Youmin Zhang, "A Review on Fault-Tolerant Control for Unmanned Aerial Vehicles (UAVs)", American Institute of Aeronautics and Astronautics, 2011, pp. 1–12.
10. Isermann, "Supervision, Fault- Detection and Fault-Diagnosis Methods- An Introduction", Elsevier science Ltd, 1997, pp. 639–652.
11. Hamid R. Berenji and Yan Wang, "Wavelet Neural Networks for Fault Diagnosis and Prognosis", IEEE International Conference on Fuzzy Systems, 2006, pp. 1334–1339.
12. Ravi Kapadia, Greg Stanley, Mark Walker, "Real World Model-based Fault Management", pp. 1–8.
13. Georg Jager, Sebastian Zug, TinoBrade, Ana-Maria Cretu, Christian Moewes: "Assessing neural networks for sensor fault detection"
14. Samy. I. Postlethwaite, D.-W. Gu, I.S. Fan "Detection of multiple sensor faults using neural networks- demonstrated on an unmanned air vehicle (UAV) model"
15. Gurcan Lokman, Guray Yilmaz "A new method for Anomaly Detection and Target Recognition", International Conference on Unmanned Aircraft Systems (ICUAS), Orlando, FL, USA, 2014. pp. 77–583, 2014