Fault Detection & Diagnosis for Small UAVs via Machine Learning

Elgiz Baskaya
ENGIE Ineo - Groupe ADP -
SAFRAN RPAS Chair
UAS Lab, ENAC
Toulouse, FRANCE
Email: elgiz.baskaya@enac.fr

Murat Bronz
UAS Lab, ENAC
Toulouse, FRANCE
Email: murat.bronz@enac.fr

Daniel Delahaye
Laboratoire MAIAA, ENAC
Toulouse, FRANCE
Email: deniel.delahaye@enac.fr

Fault can be detected using additional hardware which is known as hardware redundancy or can be detected using algorithms known as analytical redundancy

***Abstract*—The new era of small UAVs necessitates intelligent approaches towards the issue of fault diagnosis to ensure a safe flight. A recent attempt to accommodate quite a number of UAVs in the airspace requires to assure a safety level. The hardware limitations for these small vehicles point the utilization of analytical redundancy rather than the usual practice of hardware redundancy in the conventional flights. In the course of this study, fault detection and diagnosis for aircraft is reviewed. An approach of implementing machine learning practices to diagnose faults on a small fixed-wing is selected. The selection criteria behind is that, data- driven fault diagnosis enables avoiding the burden of accurate modeling needed in model-based fault diagnosis.**

In this study, first, a model of an aircraft is simulated. This model is not used for the design of Fault Detection and Diagnosis (FDD) algorithms, but instead utilized to generate data and test the designed algorithms. The measurements are simulated using the statistics of the hardware in the house. Simulated data is opted instead of flight data to isolate the probable effects of the controller on the diagnosis, which will complicate this preliminary study on FDD for drones.

A supervised classification method, SVM (Support Vector Machines) is used to classify the faulty and nominal flight conditions. The features selected are the gyro and accelerometer measurements. The fault considered is loss of effectiveness in the control surfaces of the drone. Principle component analysis is used to investigate the data by reducing the feature space dimension. The training is held offline due to the need of labeled data. The results show that for simulated measurements, SVM gives very accurate results on the classification of loss of effectiveness fault on the control surfaces.

I. Introduction

The cost effectiveness and reachability of COTS elements, shrinking size of electronics serve as a perfect environment for small flying vehicles to emerge. This accelerating trend towards small but capable flying vehicles is pushing the limits of both hardware and software potentials of industry and academia. Increasing usage of these vehicles for a variety of missions pushes a further liability to secure the flight.

To achieve a safe flight is not an easy task considering the unknowns of the systems hardware, environment and possible system faults and failures to emerge. Also, increasing demand on cost effective systems, resulting in the smaller sensors and actuators with less accuracy, impose the software to achieve even more. The expectation that UAVs should be less expensive than their manned counterparts might have a hit on reliability of the system. Cost saving measures other than the need to support a pilot/crew on board or decrement in size would probably lead to decrease in system reliability.

Systems are often susceptible to faults of different nature. Existing irregularities in sensors, actuators, or controller could be amplified due to the control system design and lead to failures. A fault could be hidden thanks to the control action (1).

The widely used method to increase reliability is to use more reliable components and/or hardware redundancy. Both requires an increase in the cost of the UAS conflicting one of the main reasons of UAS design itself band consumer expectations (2). To offer solutions for all different foreseen categories of airspace, a variety of approaches should be considered. While hardware redundancy could cope with the failure situations of UAVs in the certified airspace, it may not be suitable for UAVs in open or some subsets of specific categories due to budget constraints. Analytical redundancy is another solution, may be not as effective and simple as hardware redundancy, but relies on the design of intelligent methods to utilize every bit of information on board aircraft wisely to deal with the instances.

There are three approaches to achieve safe FTC in standard flight conventions. First one is the fail operational systems which are made insensitive to any single point component failure. The second approach is the fail safe systems where a controlled shut down to a safe state is practiced whenever a critical fault is pointed out by a sensor. The level of degradation assures to switch to robust (alternate) or direct (minimal level of stability augmentation independent of the nature of the fault) mode. Switching from nominal mode to the robust and direct modes leads to a decrease in the available GNC functions. This causes a degradation in ease of piloting. And also some optimality conditions could have

GNC - Guidance Navigation and Control

978-1-5386-0365-9/17/$31.00 ©2017 IEEE

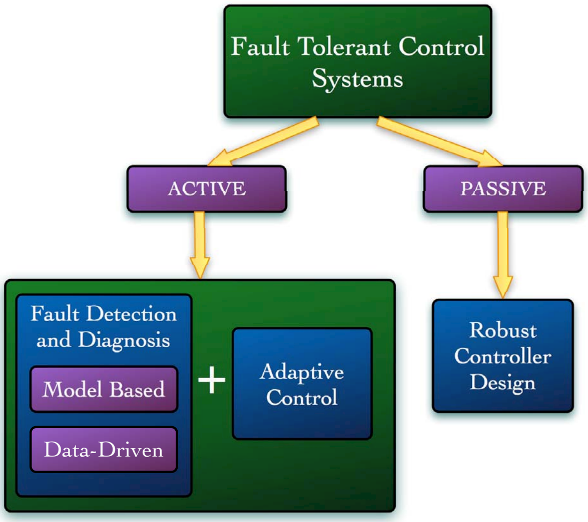

Fig. 1. Variations of fault tolerant control systems

been compromised. The third approach is fault tolerant control systems in which redundancy in the plant and the automation system is employed to design software that monitors the components and takes in action whenever needed. The strategy is most probably to try to keep plant availability and accept reduced performance (3).

Fault Tolerant Control System

II. Methods for FTCS

A common categorization of FTCS is passive and active FTCS. In passive FTCS, the flight controller is designed in such a way to accommodate not only the disturbances but also the faults. Active FTCS first distinguishes the fault via fault detection and diagnosis module and then switch between the designed controllers specific to the fault case or design a new one online (2). While active FTCS requires more tools to handle faults as seen in Fig. 1, for faults not predicted and not counted for during the design of the robust controller, this method most probably fails.

Even with a long list of available methods, aerospace industry has not implemented FTC widely, except some space systems, due to the evolving nature of the methods, the tricks coming with the nonlinear nature of the problem, design complexity and high possibility of wrong alarms in case of large disturbances and/or modeling uncertainties. So the already carried reliability measures concerning the hardware redundancy is now the preferred way because of its ease and maturity being implemented on various critical missions with considering human lives.

III. Fault Detection and Diagnosis

FDD is handled in two main steps; fault detection and fault diagnosis. Fault diagnosis encapsulates fault isolation and fault identification. The methods for detection and diagnosis are investigated for their frequency of utilization separately for sensor, actuator, process and controller faults in (4). FDD should not only be sensitive to the faults but also robust to the model uncertainties and external disturbances.

Two distinct options to proceed in analytical redundancy are the model based approaches and data-driven approaches. They form the two ends of a continuous solution set line, so utilizing them in a combination might end up with better solutions. Model based fault diagnosis highlights the components of a system and the connections in-between, and their corresponding fault modes. Data driven fault diagnosis rely on the observational data and prefers dense, redundant and with a frequency larger than the failure rate.

A. Model Based

In model based approaches, relations between measurements and estimated states are exploited to detect possible dysfunction. The most common ways to implement a model based approach is to estimate the states, estimate the model parameters, or parity-space. The accuracy of the results depend on the type of faults (additive or multiplicative). Additive faults affects the variables of the process by a summation whereas the multiplicative faults by a multiplication. When only output signal can be measured, signal model based methods can be employed for fault detection such as Bandpass filters, Spectral analysis(FFT) and maximum entropy estimation. For the case, both the input and output signals are available, the utilized methods for fault detection are called the process based methods: State and output observers(estimators), Parity equations and Identification and parameter estimation. They generate residuals for state variables or output variables. When previous works investigated, it is concluded that the most widely used technique for sensor and actuator faults is the state and output observers (estimators) and for process faults, identification and parameter estimation (4).

The output of the model based fault detection methods is the stochastic behaviour with mean values and variances. With the use of change detection methods, deviations from the normal behavior can be detected. For that purpose, three available methods considered are, mean and variance estimation, likelihood-ratio-test and Bayes decision, run-sum test and two-probe t-test. Fault detection is only supported by simple threshold logic or hypothesis testing in most of the applications (4).

A bunch of studies discovers the band of different approaches for model-based fault detection. Detecting sensor and actuator faults via state estimation, utilizing an EKF is applied to a F-16 model in (5). Parameter identification via H_∞ filter is used to indicate icing in (6).

A drawback of model-based approaches is that they require accurate model of the aircraft for successful detection. In a small UAV system susceptible to various uncertainties/disturbances and most of the cases does not have an accurate model, leading a model-based approach might fail. And also, a mathematical model of a UAV is constructed within the flight envelope, and does not necessarily describe the possible dynamics invoked by a failure on board.

u = Euc + uf

A way to handle that is to offer solutions to cope with the uncertainties. A fairly old study in 1984, investigates the design problem FDI systems robust to uncertainties within the models. One of the two steps of FDI, two steps being the residual generation and decision-making, is targeted. They offer to handle model uncertainties, by designing a robust residual generation process (7). Another study deals with model uncertainties by determining the threshold of the residual in a novel way with an application to detect aileron actuator fault (8). (9) utilize two cascade sliding mode observers state estimation and fault detection to guarantee staying in sliding manifold in the presence of unknown disturbances and faults.

B. Machine Learning

Model-based approaches had various successful applications until now, most of them assuming accurate model is available on board. With the new era of UAVs, the airspace is expected to be populated by an abrupt increase in the number of UAVs. The variety of UAVs, expense of accurate modeling practices, the difficulty in modeling the behavior of UAV in case of failures, call for alternative approaches for the quite challenging problem of FDD. The increased efficiency of sensors on board, the increase in the computational capabilities of autopilot processors, and the advances in machine learning techniques in the last decade may offer efficient data-driven solutions to FDD.

In data driven methods, a detailed knowledge about the internal dynamics of the system is not necessary. The data available is the source of information with regard to the behavior of the system. Supervised learning, which requires to label the fault cases previously in the training data, is usually utilized for data-centric inference of causes. In case of an unlabeled fault, the result is expected as a probability distribution of the available normal modes, identified fault labels and a probable unknown fault. What is needed at that point is to first detect and localize the fault and then to consult domain experts for labeling for further integration of this fault into the diagnosis scheme (10).

Amidst data driven methods for FDD, such as Neural Networks (11) and Principal Component Qnalysis (PCA) (12), Support Vector Machines (SVM) appear more recently in the literature. (13) argues artificial intelligence methods for fault detection of complex systems. Comparison between PCA and model based stochastic parity space approaches is given in (14). In (15), the authors argues to use dynamic PCA since UAV flight controls is a dynamic system itself and DPCA can reflect unknown disturbances, while model-based approaches can only model typical disturbance.

SVM is introduced in 1964 in the statistical learning theory domain and relies on structural risk minimization principle (16). Although the theory has old roots, its application to classification as a machine learning algorithm is recent and originally offer solutions for two-class classification (17; 18). SVM's first application as a classifier was mainly on object classification in images and followed by fault detection lately. The use of SVM on fault detection has gained popularity thanks to its improvement in accuracy of detection (19). Application of SVM on fault detection is mostly held in mechanical machinery, such as roller bearings, gear box, turbo pump rotor and sometimes other systems; semi-conductors, refrigeration systems and chemical processes. Its application on complex systems has not been very widely adopted yet and forms the basis of study for our research.

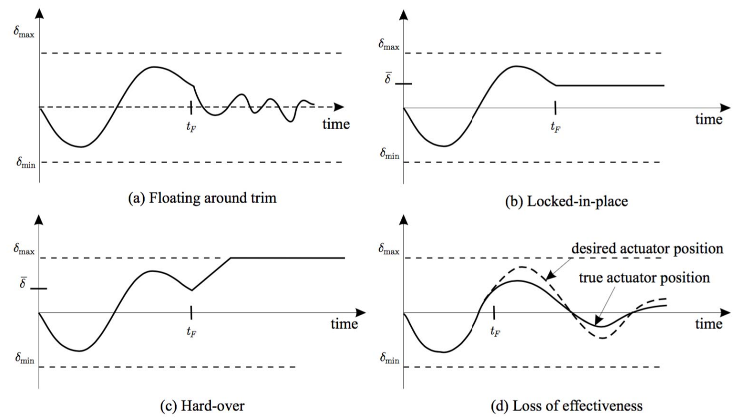

Fig. 2. Common actuator faults (1)

IV. System Modeling

FDI - Fault Detection and Isolation

In this study, first, a model of an aircraft is simulated. This model, will not be used for the design of FDI algorithms, but instead will be utilized to test them. Nonlinear aircraft flight dynamics for translational and attitude motion can be given as a system of first order partial differential equations

$$\dot{\boldsymbol{x}}_{NED} = \boldsymbol{C}_b^n \boldsymbol{v}^b \tag{1}$$

$$\dot{\boldsymbol{v}}^b = \frac{1}{m}\left[m\boldsymbol{g}^b + \boldsymbol{F}_t^b + \boldsymbol{F}_a^b\right] - \boldsymbol{\omega}_{b/i}^{b\ \times} \boldsymbol{v}^b \tag{2}$$

$$\dot{q}_0 = -\frac{1}{2}\boldsymbol{q}_\nu^T \boldsymbol{\omega}_{b/i}^b \tag{3}$$

$$\dot{\boldsymbol{q}}_\nu = \frac{1}{2}\left(\boldsymbol{q}_\nu^\times + q_0 \boldsymbol{I}_3\right)\boldsymbol{\omega}_{b/i}^b \tag{4}$$

$$\boldsymbol{J}\dot{\boldsymbol{\omega}}_{b/i}^b = \boldsymbol{M} - \boldsymbol{\omega}_{b/i}^{b\ \times} \boldsymbol{J}\boldsymbol{\omega}_{b/i}^b \tag{5}$$

where $\boldsymbol{x}_{NED} \in \mathbb{R}^3$ is the position of the center of mass of UAV with respect to inertial frame $\mathcal{I}$ expressed in the body frame $\mathcal{B}$, $\boldsymbol{v}^b$ is the velocity of the center of mass of UAV with to $\mathcal{I}$ expressed in $\mathcal{B}$, $\boldsymbol{q} = [q_0, \boldsymbol{q}_v^T]^T \in \mathbb{R}^3 \times \mathbb{R}$ is the unit quaternion representing the attitude of the body frame $\mathcal{B}$ with respect to inertial frame $\mathcal{I}$ expressed in the body frame $\mathcal{B}$, $\boldsymbol{J} \in \mathbb{R}^{3\times 3}$ is the positive definite inertia matrix of the drone, $\boldsymbol{M} \in \mathbb{R}^3$ represents the moments acting on the drone. The notation $\boldsymbol{x}^\times$ for a vector $x = [x_1 \quad x_2 \quad x_3]^{\mathrm{T}}$ represents the skew-symmetric matrix

$$\boldsymbol{x}^\times = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \tag{6}$$

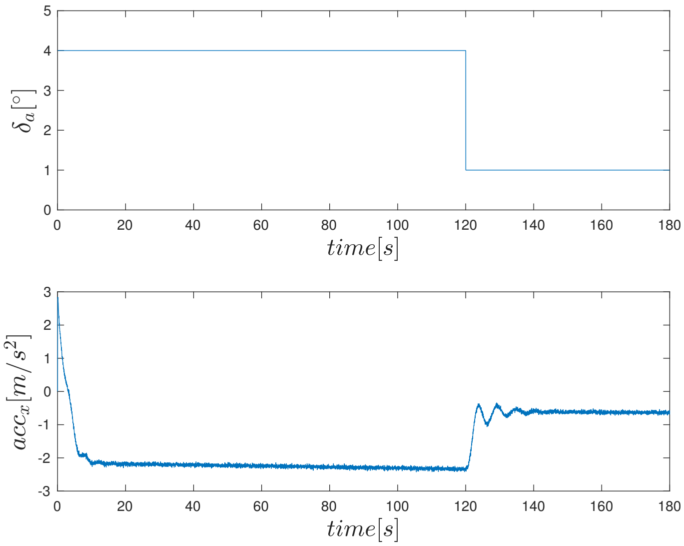

Fig. 3. Loss of effectiveness fault simulation in aileron command and corresponding accelerometer x axis measurement

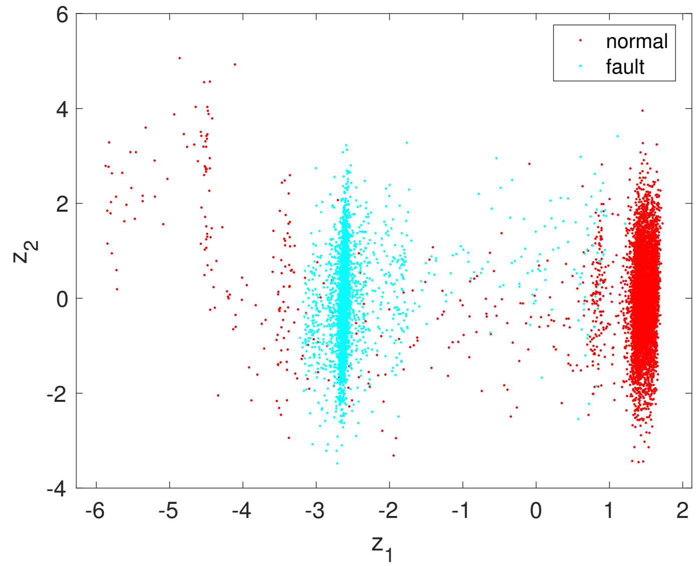

Fig. 4. Principal component analysis for visualization of faulty and normal data in reduced dimensional feature space

Aileron controls the roll of an airplane and elevator controls the pitch (up and down) motion
Roll means rotation about the longitudinal axis.

The stability and aerodynamic force coefficients are generated by AVL. The input vector can be written as $\boldsymbol{u}(t) \in \mathbb{R}^3$

$$\boldsymbol{u}(t) = \begin{bmatrix}\delta_a & \delta_e & n\end{bmatrix}^{\mathrm{T}} \tag{7}$$

Here δ_a aileron deflection angle in degrees, δ_e elevator deflection angle in degrees, n engine speed in rev/s.

When the actuators are healthy, actual control input signal will be equal to the given input signal. In case of a fault the actual signal can be modeled as

$$\boldsymbol{u}(t) = \boldsymbol{E}\boldsymbol{u}_c + u_f \tag{8}$$

where $\boldsymbol{u}_c$ is the desired control signal, $E = diag(e_1, e_2, e_3)$ is the effectiveness of the actuators where $0 \leq e_i \leq 1$ with $(i = 1, 2, 3)$ and u_f additive actuator fault. This model makes it possible to simulate all four types of actuator faults shown in Fig. 2. Most of the FDI algorithms are implemented to open-loop systems, ignoring the probable influences of the controller might cause on the detection performance (20). Here the system is open-loop as well. Further implementation of a controller is foreseen to understand the effect of the selected controllers. So we follow a step by step approach and hope to end with a more realistic case, in which real flight data is utilized and diagnosis is achieved online aside a functioning controller.

Since it is not possible to see all features, we take advantage of the dimensionality reduction technique called Principle Component Analysis (PCA) for visualization. Here what we do is to map the feature vector, $\boldsymbol{x} \in \mathbb{R}^n$ to a lower dimensional space where the new feature set will be represented by $\boldsymbol{z} \in \mathbb{R}^k$. Fig. 4 shows the resulted most significant elements for a mapped feature space from six dimensional feature vector to two.

V. Classification of Fault via SVM

Helps in generalisation

SVM is a relatively new approach for classification offering better generalization property thanks to its foundations on the structural risk minimization principle (21; 22) while other classifiers usually only minimizes the empirical risk. This advances the capacity of generalization even with a small number of instances by reducing the risk of overfitting for a nicely tuned parameters setting. It can be applied to nonlinear systems and problems offering a vast number of features. Furthermore, taking advantage of convex optimization problems in the solution of SVM models, another attractive reason to use SVM rises as avoidance of global minimas, while Neural Networks is inherently prone to local minimas.

The idea behind SVM is to find an optimal hyperplane that will linearly separate the classes. This is achieved with the introduction of maximum margin concept which is the distance in between the boundaries when they are extended until hitting the first data point as in Fig. 5. The points closest to the hyperplane (decision boundary) are called the support vectors and are the representatives of the data sets to be used for the decision process. This helps to decrease the data to handle abruptly, enhancing the ability to cope with the curse of dimensionality and reducing the computational complexity.

SVM has other tricks to deal with not linearly seperable problems such as using kernels to map data into higher dimensional feature spaces where they can be separated with a linear hyperplane.

A binary classifier is used in this work to classify two classes, faulty and nominal. The fault considered in this study is the loss of effectiveness of the control surfaces. SVM being a supervised classification algorithm has two main phases as shown in Fig. 6. In the training phase, the model is learned as a fit to the labeled data that is fed to the SVM algorithm. This

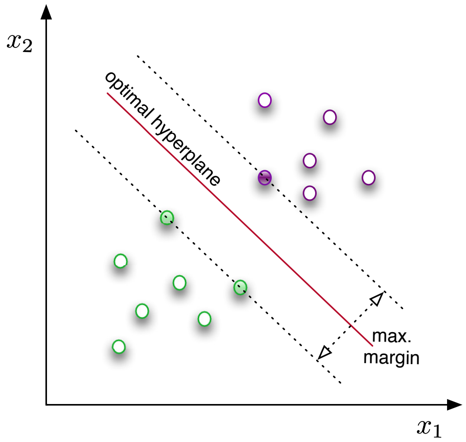

Fig. 5. SVM working principle

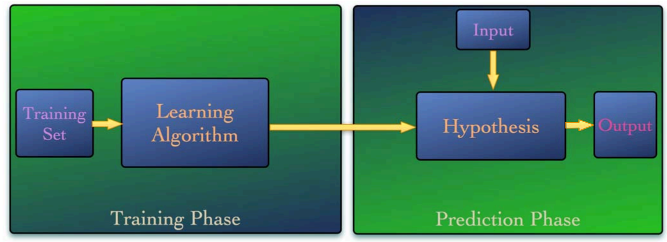

Fig. 6. Supervised learning basics

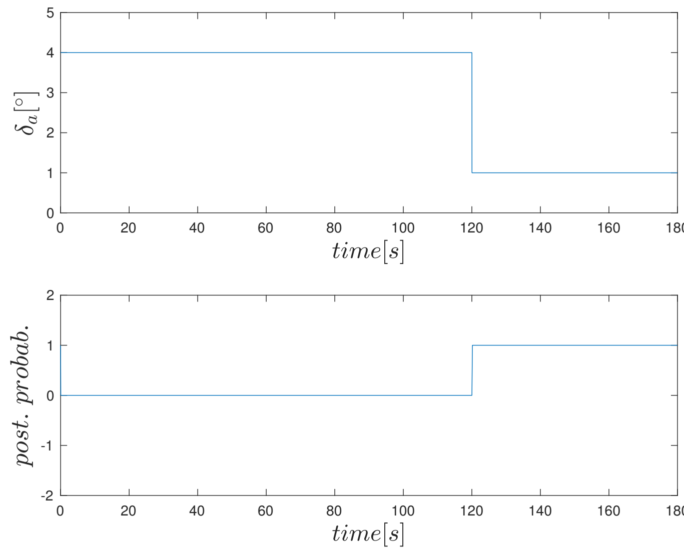

Fig. 7. Posterior probability of loss in effectiveness fault for test set when a fault is injected at t = 120s.

phase is usually followed with a tuning phase where some of the parameters of SVM is changed and results are compared to have the best fit via cross validation to avoid overfitting. The last phase is the prediction, where for a new instance the classifier predicts if it corresponds to a faulty or nominal condition.

Training data is comprised of labeled data where the label can belong to one of two possible cases. This data set is saved in $\boldsymbol{X} \in \mathbb{R}^{m \times n}$ where m, n correspond to number of instances and features respectively. The label information corresponding to the measurement instances is also fed to the SVM algorithm during the training phase as output vector $\boldsymbol{y} \in \{-1, 1\}$. The aim of SVM is to find an optimal hyperplane maximizing the margin by solving the optimization problem for non-linearly separable datasets

$$min_{\gamma,\omega,b} \quad \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{m}\xi_i \tag{9}$$

$$s.t. \quad y^i(\omega^T x^(i) + b) \geq 1 - \xi_i, \;\; i = 1, \cdots, m \tag{10}$$

$$\xi_i \geq 0, \;\; i = 1, \cdots, m \tag{11}$$

$$\tag{12}$$

To avoiding overfitting, which is the main problem of parametric discrimination approaches such as neural networks, parameter C is tuned to result in the optimal fit for the cross validation set. The data set available is first divided to two portions with a percentage of %20, %80 where the bigger chunk is the training set and the remaining is the test set. Further, the training set is divided as cross-validation and training sets. The idea to split data is to avoid overfitting. Overfitting means that the models trained being very accurate fit for the data they are trained to but fail to generalize with new inputs resulting in bad prediction performance for the new data. To assess the performance of the classifier trained with the training data is tuned to give a better performance with the cross validation data. And then the final ability of the classifier is tested on the test set. This parameter also tuned for the outliers to generalize the distribution of the data rather than resulting in fine fits for each individual data in the training set. With a satisfactory result of the training & tuning is followed by the prediction where the classifier predicts if the new measurement data belongs to the faulty or nominal class. The output of the SVM classification is not the probability that the new measurement belongs to one class as is in the traditional classification problems, but directly the class information it belongs to. For investigating the performance of the classifier on the test set, a method (23) is used to calculate the posterior probabilities giving the probability that the new measurements belongs to faulty mode. Results shows as in Fig. 7 that prober tuning achieves very accurate and instant detection for the drone fault.

VI. Conclusion

Integration of drones into airspace needs the introduction of indigenous designs that will serve safe solutions for drones. One of the aspects of the problem is to assure a safe flight by designing fault detection and diagnosis with cheaper avionics common in a vast number of drones projected. This work aims to design a classifier via SVM to solve FDD of drones with actuator faults. This problem possess various challenges. This work focuses on a loss of effectiveness fault which is more difficult than a stuck fault to diagnose, but easier to mitigate.

A model of a MAKO UAV is simulated to generate data and test the designed algorithms. The simulated data of gyro and accelerometer measurements are given to classifier to train for the two class labeled data set. A supervised classification method, SVM (Support Vector Machines) is used to classify the faulty and nominal flight conditions. Principle component analysis is used to investigate the data by reducing the feature space dimension. The training is held offline due to the need of labeled data but prediction is envisioned be held real time. The results show that for simulated measurements, SVM gives very accurate results on the classification of loss of effectiveness fault on the control surfaces.

Further study is envisaged to deal with the controller diagnosis interaction and classification of multiple faults. Also discussion of SVM for online training might be addressed since SVM is in need for labeled data which requires generating the labeled data during flight.

Acknowledgment

This work was supported by the ENGIE Ineo - Groupe ADP - SAFRAN RPAS Chair.

References

[1] G. J. Ducard, *Fault-tolerant flight control and guidance systems: Practical methods for small unmanned aerial vehicles*. Springer Science & Business Media, 2009.

[2] P. Angelov, *Sense and avoid in UAS: research and applications*. John Wiley & Sons, 2012.

[3] M. Blanke, C. W. Frei, F. Kraus, R. J. Patton, and M. Staroswiecki, "What is fault-tolerant control," in *Preprints of 4th IFAC Symposium on Fault Detection Supervision ans Safety for Technical Processes, SAFEPROCESS*, 2000, pp. 40–51.

[4] R. Isermann and P. Ballé, "Trends in the application of model-based fault detection and diagnosis of technical processes," *Control engineering practice*, vol. 5, no. 5, pp. 709–719, 1997.

[5] C. Hajiyev and F. Caliskan, "Sensor and control surface/actuator failure detection and isolation applied to f-16 flight dynamic," *Aircraft Engineering and aerospace technology*, vol. 77, no. 2, pp. 152–160, 2005.

[6] J. W. Melody, T. Hillbrand, T. Başar, and W. R. Perkins, "H_∞ parameter identification for inflight detection of aircraft icing: The time-varying case," *Control Engineering Practice*, vol. 9, no. 12, pp. 1327–1335, 2001.

[7] E. Chow and A. Willsky, "Analytical redundancy and the design of robust failure detection systems," *IEEE Transactions on Automatic control*, vol. 29, no. 7, pp. 603–614, 1984.

[8] H. Rotstein, R. Ingvalson, T. Keviczky, and G. J. Balas, "Fault-detection design for uninhabited aerial vehicles," *Journal of guidance, control, and dynamics*, vol. 29, no. 5, pp. 1051–1060, 2006.

[9] R. Sharma and M. Aldeen, "Fault detection in nonlinear systems with unknown inputs using sliding mode observer," in *2007 American Control Conference*. IEEE, 2007, pp. 432–437.

[10] J. Stutz, "On data-centric diagnosis of aircraft systems," *IEEE Transactions on Systems, Man and Cybernetics*, 2010.

[11] M. Schlechtingen and I. F. Santos, "Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection," *Mechanical systems and signal processing*, vol. 25, no. 5, pp. 1849–1875, 2011.

[12] X. Sun, H. J. Marquez, T. Chen, and M. Riaz, "An improved pca method with application to boiler leak detection," *ISA transactions*, vol. 44, no. 3, pp. 379–397, 2005.

[13] W.-h. Gui and X.-y. Liu, "Fault diagnosis technologies based on artificial intelligence for complex process," *Basic Automation*, vol. 4, p. 000, 2002.

[14] A. Hagenblad, F. Gustafsson, and I. Klein, "A comparison of two methods for stochastic fault detection: the parity space approach and principal component analysis," 2004.

[15] M. Li, G. Li, and M. Zhong, "A data driven fault detection and isolation scheme for uav flight control system," in *Control Conference (CCC), 2016 35th Chinese*. TCCT, 2016, pp. 6778–6783.

[16] V. Vapnik and A. Chervonenkis, "A note on one class of perceptrons," *Automation and remote control*, vol. 25, no. 1, p. 103, 1964.

[17] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 1992, pp. 144–152.

[18] V. Vapnik, "The nature of statistical learning theory springer new york google scholar," 1995.

[19] N. Laouti, N. Sheibat-Othman, and S. Othman, "Support vector machines for fault detection in wind turbines," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 7067–7072, 2011.

[20] R. Pandita, J. Bokor, and G. Balas, "Closed-loop performance metrics for fault detection and isolation filter and controller interaction," *International Journal of Robust and Nonlinear Control*, vol. 23, no. 4, pp. 419–438, 2013.

[21] S. R. Gunn *et al.*, "Support vector machines for classification and regression," *ISIS technical report*, vol. 14, pp. 85–86, 1998.

[22] S. Yin, X. Gao, H. R. Karimi, and X. Zhu, "Study on support vector machine-based fault detection in tennessee eastman process," in *Abstract and Applied Analysis*, vol. 2014. Hindawi Publishing Corporation, 2014.

[23] J. Platt *et al.*, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in large margin classifiers*, vol. 10, no. 3, pp. 61–74, 1999.