

# Neural Network Based Dynamic Model and Gust Identification System for the Jetstream G-NFLA

---

**Aristeidis Antonakis, Mudassir Lone and Alastair Cooke**

Dynamics, Simulation and Control Group,

Centre for Aeronautics

Cranfield University

Cranfield MK43 0AL

## **Keywords:**

Artificial Neural Networks , Flight Testing , System Identification , Gust Identification , Hybrid Identification

## **1. Abstract**

Artificial Neural Networks are an established technique for constructing non-linear models of Multi-Input-Multi-Output systems based on sets of observations. In terms of aerospace vehicle modeling, however, these are currently restricted to either unmanned applications or simulations, despite the fact that large amounts of flight data are typically recorded and kept for reasons of safety and maintenance. In this paper, a methodology for constructing practical models of aerospace vehicles based on available flight data recordings from the vehicles' operational use is proposed and applied on the Jetstream G-NFLA aircraft. This includes a data analysis procedure to assess the suitability of the available flight databases and a NN-based approach for modeling. In this context, a database of recorded landings of the Jetstream G-NFLA, normally kept as part of a routine maintenance procedure, is used to form training datasets for two separate applications. A NN-based longitudinal dynamic model and gust identification system are constructed and tested against real flight data. Results indicate that in both cases, the resulting models' predictions achieve a level of accuracy that allows them to be used as a basis for practical real-world applications.

## **2. Nomenclature**

$a_x$       Body Longitudinal Acceleration

$a_y$       Body Lateral Acceleration

$a_z$	Body Vertical Acceleration
$dx$	CG position x-coordinate about Body Axes
$dy$	CG position y-coordinate about Body Axes
$dz$	CG position z-coordinate about Body Axes
$H$	Pressure Altitude
$H$	GPS Altitude
$L$	Rolling Moment
$M$	Pitching Moment
$m$	Mass
$N$	Yawing Moment
$P$	Body Roll rate
$P_\infty$	Ambient Pressure
$Q$	Body Pitch Rate
$Q_F$	Fuel Quantity
$R$	Body Yaw Rate
$T$	Temperature
$U$	Body Forward Velocity
$V$	Body Lateral Velocity
$V_{TAS}$	True AirSpeed
$W$	Body Vertical Velocity
$x_{CG}$	CG position about the longitudinal axis
$X$	Body Longitudinal Force
$X_{a\&p}$	Aero-propulsive Longitudinal Force
$Y$	Body Lateral Force
$Y_{a\&p}$	Aero-propulsive Lateral Force
$Z$	Body Vertical Force
$Z_{a\&p}$	Aero-propulsive Vertical Force
$\alpha$	Angle of Attack
$\beta$	Sideslip Angle
$\gamma$	Flight Path Angle
$\eta$	Elevator Deflection

$\theta$	Pitch Angle
$\zeta$	Rudder Deflection
$\xi$	Aileron Deflection
$\sigma_m$	Model Uncertainty
$\sigma_p$	Prediction Variance
$\sigma_\varepsilon$	Noise Variance
$\tau$	Throttle
$\varphi$	Roll Angle
$\psi$	Yaw Angle

### 3. List of abbreviations

ANN	Artificial Neural Network
AUM	All-Up Mass
DoF	Degrees of Freedom
EM	Empty Mass
GPS	Global Positioning System
MIMO	Multi-Input Multi-Output
NN	Neural Network
PC	Principal Component
PCA	Principal Component Analysis
PI	Prediction Intervals
PICP	Prediction Intervals' Corrected Percentage
RBF	Radial-Basis Function
RBFN	Radial-Basis Function Network
RLG	Ring Laser Gyro
RMS	Root Mean Square
SL	Sea Level

### 4. Introduction

Traditionally, modelling of aerospace vehicles has been associated with the study of linearized versions of the physical laws that govern the vehicles' dynamics. Although this approach has

proved to be widely effective, it also subject to a series of well-understood limitations, especially in applications where the knowledge of the underlying dynamic structure is limited. Typical examples are aircraft with extended flight envelopes, unconventional configurations or of a very small size.

Artificial Neural Networks (ANNs) offer an alternative solution to this problem. These are computational models that mimic the structure and operation of the biological brain. These have been shown to be capable of non-linear real-time modelling of Multi-Input-Multi-Output (MIMO) systems without requiring any assumptions on their internal structure. Trained on a database of observations, a NN can effectively replicate a system's response with an accuracy which is primarily defined by the quality and quantity of the training data.

During the last decades, exploiting their inherent ability to model non-linear systems, NNs have been extensively used in control engineering for both system identification and controller design (1) (2). More recently, the increased demand for UAV platforms has revived interest in NN-based aerospace modelling and control (3) (4) (5) (6): as a result of their reduced size and extended flight envelopes, these types of aircraft show increased sensitivity to atmospheric variations and benefit from the adaptive characteristics of NNs. NASA have also evaluated NN modelling and control techniques to full-scale aircraft with the F-15 Intelligent Flight Control System (IFCS) project (7) (2).

Recent research has attempted to expand NN applications to the study and alleviation of the effects of atmospheric turbulence on aircraft dynamic response. In this context, a number of NN-based wind gust load identification/alleviation system designs have been proposed (8) (9) (10). Their effectiveness, however, has currently only been demonstrated in a simulation environment.

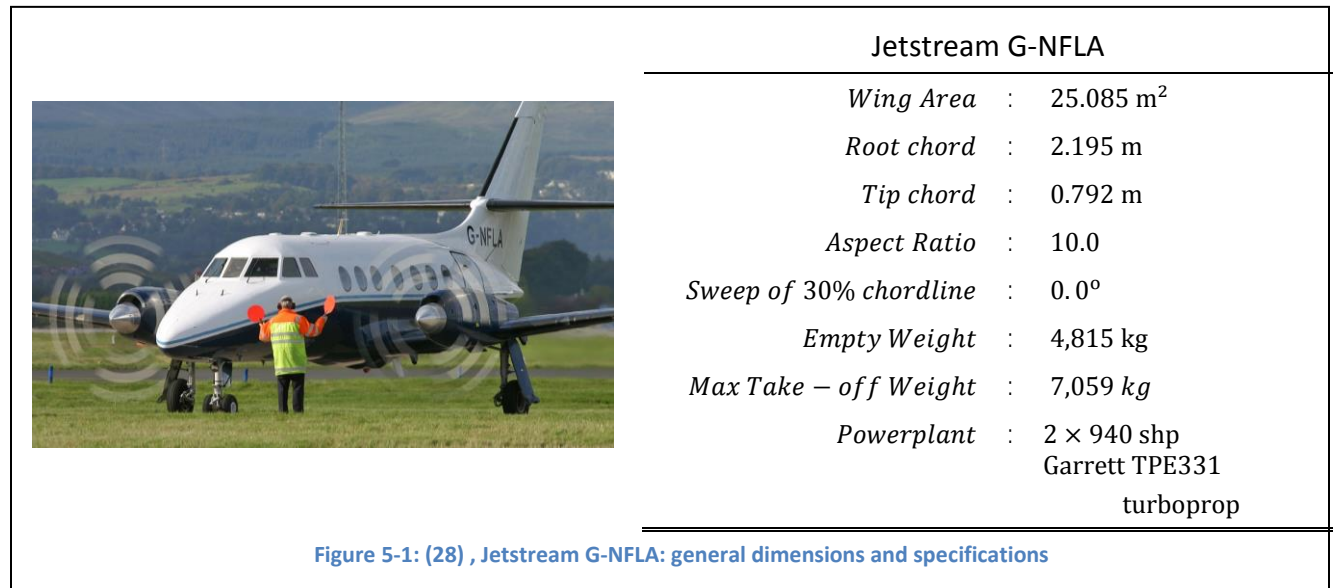
In this paper, a NN-based dynamic modelling methodology for the Cranfield University's Jetstream G-NFLA aircraft is presented as a fast and cost-effective means of building an accurate model for an aircraft using available flight data from its operational use. The flight database used was selected as representative of the datasets that are typically available to aircraft operators. As a result, all training data in this project were sourced from a set of 60 landings of the aircraft, recorded as part of routine maintenance procedures-no dedicated flight testing was conducted. To further demonstrate the method's potential, a NN-based gust identification system is also proposed and assessed.

## **5. The aircraft**

The BAE Systems Jetstream 31 is an 18-seat, twin-engine turboprop aircraft first flown on 28 March 1980. The University's Jetstream G-NFLA aircraft is operated as a Flying Laboratory and

as a result, has been modified to accommodate additional measuring equipment. This has imposed slight changes to the aircraft's empty weight and mass distribution. Recordings are routinely made of the landing using data attained from a Ring Laser Gyro (RLG), nose-mounted AoA and sideslip vanes and a GPS. In addition, powerplant thrust is estimated using torque and engine speed with propeller curves from Dowty Rotol. (11)

A summary of the aircraft's dimensions and specifications is presented in Figure 4-1.



## 6. NN structure

A Radial-Basis Function Network structure was selected for this study. These are a sub-class of NNs, that use Radial Basis Functions (RBFs) as activation functions. The term RBF refers to a type of real-valued functions whose value depends exclusively on the distance from a fixed point, called the *centre*.

The first examples of work on RBFNs date back to the 1980s with contributions from Powel (12) and Broomhead & Lowe (13). A basic distinction between RBF Networks (RBFNs) and, traditional, multi-layer feed-forward networks lies in the adoption of a multi-dimensional curve-fitting approach for training, rather than the stochastic approximation methodology (14) (15) used in the latter. This has the advantage of inducing an explicit relationship between the network's parameters and the targeted output. So, network training is reduced to a linear problem with a guaranteed solution (16). This was the main reason for their selection in this study, since for other network types training is performed using iterative procedures ('backpropagation') whose convergence strongly depends on the initialization of the NN parameters. Furthermore, RBFN structures are generally simpler than other NN structures of

the same performance. In their standard form, RBFNs consist of three layers of neurons (input, hidden and output) and have shown performance standards equivalent to higher-order multi-layer structures, though typically requiring a hidden-layer structure of increased dimensionality.

RBFNs used in this study utilize Gaussian functions for network activation, using the Mahalanobis distance (17) as the distance norm to account for the unknown distribution of training samples in the data domain. In contrast with the Euclidean distance, the Mahalanobis distance is unitless and scale invariant. Distance is measured in terms of the population's standard deviation along the direction of interest. Assuming  $P = \{x_1, x_2, \dots, x_n\}$  a population of  $n$  scalars  $x_i$  with *mean*  $\mu$  and *standard deviation*  $\sigma$ , the Mahalanobis distance of point  $x_i$  from the dataset's mean is:

$$r_{p_i} = \sqrt{\frac{(x_i - \mu)^2}{\sigma^2}} \quad (6-1)$$

Expanding to multi-dimensional datasets where  $x_i$  are *vectors*:

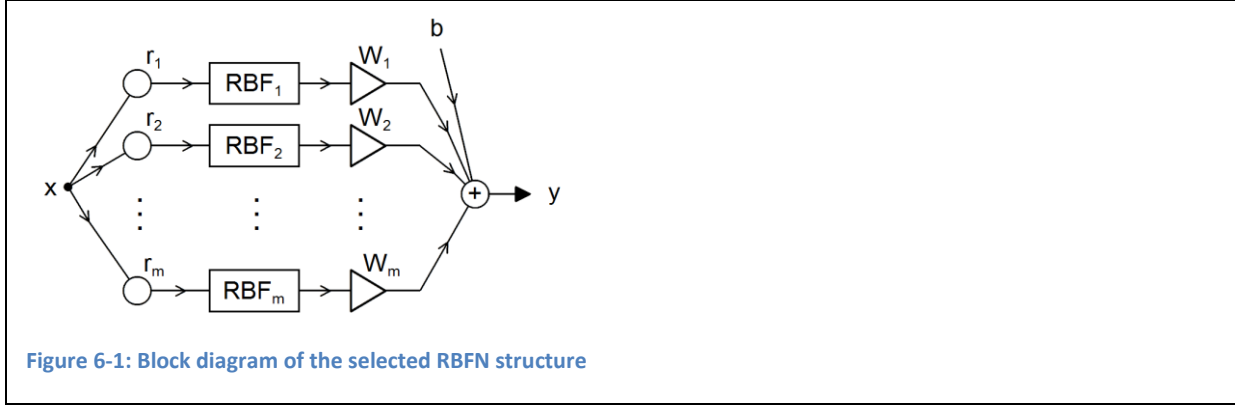
$$r_{x_i} = \sqrt{(x_i - \mu)^T R^{-1} (x_i - \mu)} \quad (6-2)$$

where  $R$  is the dataset's *covariance matrix*. For a dataset with  $m$  variables:

$$R = \begin{bmatrix} \sigma_{11}^2 & \dots & \sigma_{1m}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{m1}^2 & \dots & \sigma_{mm}^2 \end{bmatrix} \quad (6-3)$$

$$\sigma_{ij}^2 = \frac{1}{n} \sum_{k=1}^n (x_{ki} - \mu_i)(x_{kj} - \mu_j) \quad (6-4)$$

To complete the RBFN configuration, tried-and-tested schemes were employed. The average distance between RBF centers was used to scale kernel widths and a bias signal was included in the network's output layer. With respect to network training, a two-phase training scheme was selected using k-means clustering (18) for the dispersion of RBF centers. Without loss of generality, NN output was restricted to a single scalar. Vector outputs were produced by a parallel combination of multiple RBFNs. The resulting structure is shown in Figure 5-1.



## 7. Error Estimation

To obtain a complete picture of a developed model's predictive power, a measure of the accuracy of the estimations must be produced by constructing the associated *prediction intervals*. A comparison of different strategies for constructing such intervals for NNs found in the literature (19) (20) (21) (22) suggests that a combination of *maximum-likelihood* and *ensemble* techniques performs better than equivalent analytical approaches. This involves the use of bootstrapping (an ensemble technique), to estimate model uncertainty variance  $\sigma_m^2$  and a second NN, trained on squared residuals, to identify measurement noise  $\sigma_\varepsilon^2$ . Prediction variance  $\sigma_p^2$  is then defined as:

$$\sigma_p^2 = \sigma_m^2 + \sigma_\varepsilon^2 \quad (7-1)$$

Now, assuming a Gaussian error probability distribution, prediction intervals can be constructed:

$$y = \bar{y} \pm z_{\alpha/2} \cdot \sigma_p \quad (7-2)$$

where  $z_{\alpha/2}$  can be found from the normal distribution cumulative probability function, to specify a level of confidence equal to  $(1 - \alpha) \cdot 100\%$

For the applications considered in this study, it was assumed that, because of the rather simple geometry of the underlying functions, given an adequately large training dataset, model uncertainty variance  $\sigma_m^2$  is small enough to be neglected. Consequently, to reduce training time, bootstrapping was not used. Instead, a second NN was trained to directly identify noise variance  $\sigma_\varepsilon^2$ , assumed to be the only cause of prediction error. For the applications examined,  $\sigma_\varepsilon^2$  includes both atmospheric turbulence and sensor noise, which are the main factors affecting

the accuracy of in-flight measurements. 95% *prediction intervals* were constructed using the formula:

$$y = \hat{y} \pm 1.96\hat{\sigma}_\varepsilon \quad (7-3)$$

Where  $\hat{y}$  and  $\hat{\sigma}_\varepsilon$  are NN predictions.

## 8. Principal Component Analysis

Principal Component Analysis (PCA), introduced by Pearson in 1901 (23), is the oldest and most commonly used multivariate analysis technique, particularly useful in statistical function approximation applications, including NNs. PCA is used to reduce the dimensionality of the search domain (commonly caused by the inclusion of a large number of interrelated variables) while retaining the largest possible amount of variation present in the initial dataset. This is achieved through an orthogonal transformation to a new set of *uncorrelated* variables, called *principal components (PCs)*, which are ordered so that the first few retain most of the variation present in *all* of the initial variables (24). It can be shown (14) that the principal components of a dataset coincide with the *eigenvectors* of its *covariance matrix* and the respective *eigenvalues* are equal to the *variance* along each component.

As far as regression analysis is concerned, expressing a dataset in terms of its principal components is beneficial in two ways:

1. Co-linearities among variables, typically a cause of misleading or unstable predictions of the regression equation, can be identified; PCs associated with very small variances (typically, ~1% of the maximum variance) are indicators of such interconnections between the dataset's original variables. Replacing the initial set of variables with PCs, omitting those with very small variances can generally eliminate the problem.
2. The dimensionality of the dataset can be reduced. Typically, a number of components may be excluded without compromising the accuracy of the regression.

It is suggested (24) (25) that reducing the input layer's dimensionality by applying PCA to the training dataset and removing components with very small variances helps remove co-linearity problems. Joliffe (24), however, states and proves that this strategy does not apply to all cases and results strongly depend on the specific application. In the same context, Kyriacou et al (26) showed that for certain applications, omitting the PCs with the largest variances leads to improved modelling.



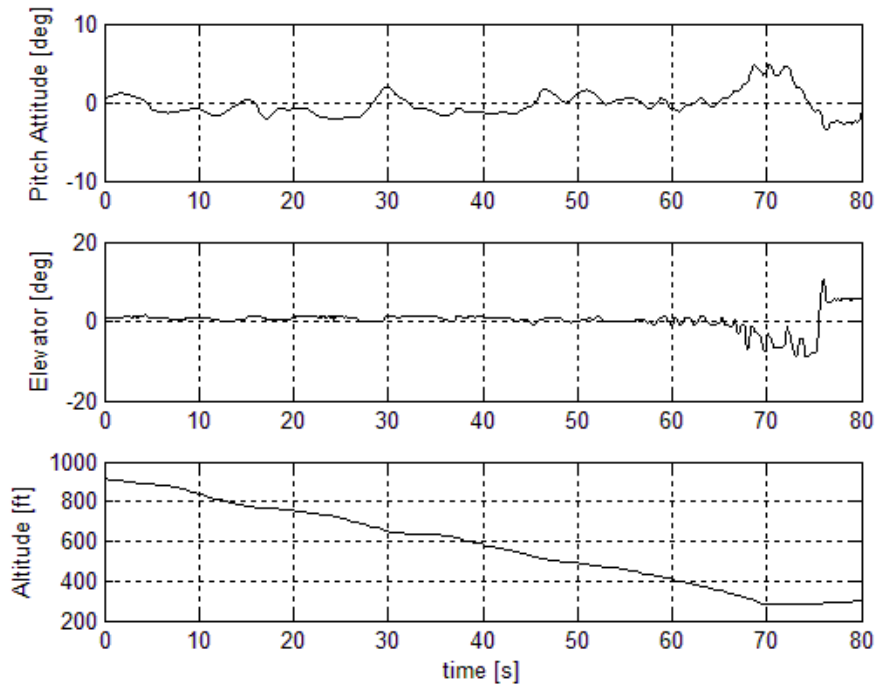


Figure 8-1: Time histories of Pitch Attitude ( $\theta$ ), Elevator ( $\eta$ ) and GPS Altitude ( $H$ ) for a typical landing

For the application considered in this paper, data comprised of a set of recorded landings that generally started at an altitude of 1,000 *ft* and ending at ground level. Flight paths were smooth and control inputs were rather small in amplitude. Figure 7-1 shows time histories of a typical landing which indicate that the maximum variance in the training dataset (comprising exclusively of recorded landings) is related to *altitude variation*, leading to the conclusion that important information are ‘hidden’ in components of smaller and unknown variance. For this reason, no PCA-driven dimensionality reduction was performed: PCA was only employed as a tool for assessing the sample distribution in the training datasets. In fact, the use of Mahalanobis distance as the RBF input is equivalent to switching to PCs for the RBFNs’ input layer.

## 9. Dynamic Modelling

A state-space approach was adopted for the dynamic modelling. In a state/space representation, a dynamic system is described by a vector of states  $x$ , a vector of control inputs  $u$  and a real function  $f$ , such that:

$$\dot{x} = f(x, u) \tag{9-1}$$

Although a feed-forward NN can be used to directly approximate function  $f$ , based on the approach found in (27) an alternative strategy was employed to reduce the dimensionality of the input space and, consequently, obtain better results for a given training dataset. Under this scope, any ‘structured’ knowledge on the aircraft was injected to the model. Aircraft dynamics were decoupled along the longitudinal and lateral/directional axes, while in the same context, the effects of known quantities such as gravity, mass and inertia were removed<sup>1</sup>. RBFN modelling was thus focused on unknown quantities, in this case the aircraft’s aerodynamic and propulsive forces and moments. A more detailed description of the selected model structure is given below.

In a typical 6-state representation of a symmetric airframe, the equations describing the aircraft’s dynamics are:

$$\begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \\ \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = M^{-1} \begin{bmatrix} X \\ Y \\ Z \\ L \\ M \\ N \end{bmatrix} + \begin{bmatrix} mRV - mQW + md_x(Q^2 + R^2) - md_yPQ - md_zPR \\ -mRU + mPW - md_xPQ + md_y(P^2 + R^2) - md_zQR \\ mQU - mPV - md_xPR - md_yQR + md_z(P^2 + Q^2) \\ -QR([I_{zz}]_b - [I_{yy}]_b) + PQ[I_{xz}]_b - (PV - QU)md_y + (RU - PW)md_z \\ -PR([I_{xx}]_b - [I_{zz}]_b) - (P^2 - R^2)[I_{xz}]_b - (QU - PV)md_x - (QW - RV)md_z \\ -PQ([I_{yy}]_b - [I_{xx}]_b) - QR[I_{xz}]_b - (RU - PW)md_x + (QW - RV)md_y \end{bmatrix} \quad (9-2)$$

Where:

$$M = \begin{bmatrix} m & 0 & 0 & 0 & md_z & -md_y \\ 0 & m & 0 & -md_z & 0 & md_x \\ 0 & 0 & m & md_y & -md_x & 0 \\ 0 & -md_z & md_y & [I_{xx}]_b & 0 & -[I_{xz}]_b \\ md_z & 0 & -md_x & 0 & [I_{yy}]_b & 0 \\ -md_y & md_x & 0 & -[I_{xz}]_b & 0 & [I_{zz}]_b \end{bmatrix} \quad (9-3)$$

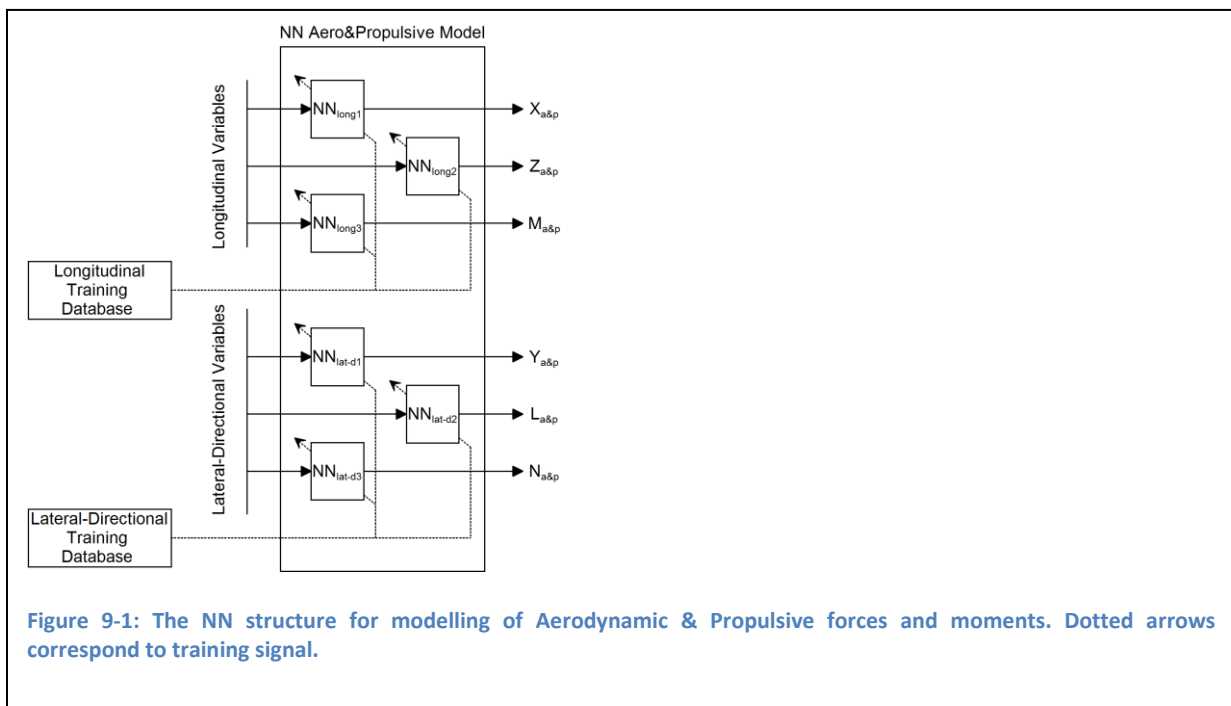
Consequently, if mass and inertia and CG position are known, the problem of dynamic modelling can be reduced to that of identifying the forces and moments acting on the airframe as a function of the flight condition.

Forces and moments acting on an aircraft in flight are of either aerodynamic/propulsive or gravitational origin. The latter are of known direction and magnitude.

<sup>1</sup> Deterministic models were employed to add gravitational components and convert the outputs to accelerations.

$$\begin{bmatrix} X \\ Y \\ Z \\ L \\ M \\ N \end{bmatrix} = \begin{bmatrix} X_{a\&p} + X_g \\ Y_{a\&p} + Y_g \\ Z_{a\&p} + Z_g \\ L_{a\&p} + L_g \\ M_{a\&p} + M_g \\ N_{a\&p} + N_g \end{bmatrix} = \begin{bmatrix} X_{a\&p} \\ Y_{a\&p} \\ Z_{a\&p} \\ L_{a\&p} \\ M_{a\&p} \\ N_{a\&p} \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -dz & dy \\ dx & 0 & -dz \\ -dy & dx & 0 \end{bmatrix} \begin{bmatrix} -mg \sin\theta \\ mg \sin\phi \cos\theta \\ mg \cos\phi \cos\theta \end{bmatrix} \quad (9-4)$$

A 6-DoF dynamic model can thus be built around the feed-forward NN structure shown in Figure 8-1. Assuming that the aircraft's longitudinal and lateral/directional dynamics are independent, the input variables can be classified so as to reduce the corresponding dimensionality of the NNs' input layers.



The complete model structure is depicted in Figure 8-2.

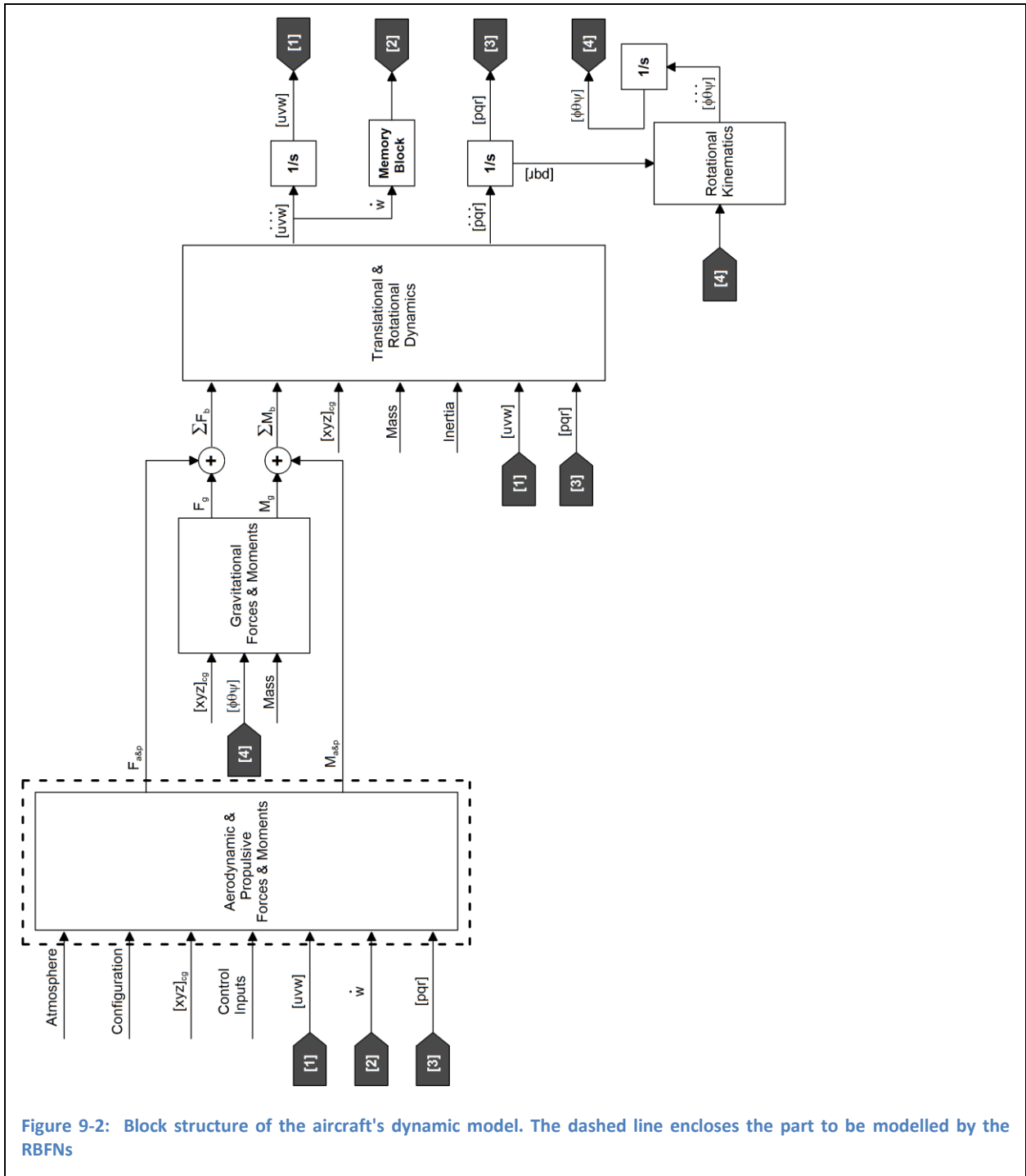


Figure 9-2: Block structure of the aircraft's dynamic model. The dashed line encloses the part to be modelled by the RBFNs

## 10. Construction of the training database

For reasons of maintenance, the approach and landing phases of all Jetstream flights are recorded. Recording is automatically initiated at an altitude of 1,000 *ft* and manually

terminated on the ground. In total, 39 parameters are recorded, at a sample rate of 25 Hz<sup>2</sup>. Among these and based on the input variables for the model of reference (27), 14 parameters were selected for use in the formation of the training database. These are listed in Table 9-1. All speeds and angles are sourced from the aircraft’s inertial system.

	Parameter	Symbol	Units	Source
1.	<i>pitch attitude</i>	$\theta$	[deg]	LRG
2.	<i>roll attitude</i>	$\varphi$	[deg]	LRG
3.	<i>Body pitch rate</i>	$Q$	[deg.s <sup>-1</sup> ]	LRG
4.	<i>Body normal acceleration</i>	$a_z$	[g]	LRG
5.	<i>Body longitudinal acceleration</i>	$a_x$	[g]	LRG
6.	<i>Flight Path Angle</i>	$\gamma$	[deg]	LRG
7.	<i>True Airspeed</i>	$V_{TAS}$	[kts]	LRG
8.	<i>Pressure Altitude</i>	$h$	[ft]	Air Data
9.	<i>Temperature</i>	$T$	[C]	Air Data
10.	<i>Elevator</i>	$\eta$	[deg]	Potentiometer
11.	<i>GPS Altitude</i>	$h_{GPS}$	[ft]	GPS
12.	<i>Port Thrust</i>	$\tau_p$	[N]	Thrust Estimation
13.	<i>Starboard Thrust</i>	$\tau_s$	[N]	Thrust Estimation
14.	<i>Fuel Quantity</i>	$Q_F$	[kg]	Aircraft

Table 10-1: Selection of data recorder parameters for the formation of the training database

Due to the lack of sideslip angle ( $\beta$ ) as a recorded parameter, modelling was restricted to the aircraft’s longitudinal dynamics. The following post-processing steps were undertaken to construct the training database:

1.  $H_{GPS}$  was used to determine, the touch-down point. All data following that were rejected.
2. Loading data and fuel quantity  $Q_F$  were used to determine all up mass, moments of inertia and CG position
3. Angular accelerations were obtained by differentiating the signals of the respective angular rates. A central-difference scheme was employed.
4. Angle of Attack (A) was calculated as follows:

<sup>2</sup> It should be noted that this sampling rate is over three times that in typical civil aircraft flight data recorders

$$\alpha = \theta - \gamma \quad (10-1)$$

5. Assuming  $V \approx 0$ , vertical speed rate  $\dot{W}$  was calculated as:

$$\dot{W} = a_z + QU \quad (10-2)$$

6. All accelerations were converted to forces and moments, subtracting gravitational effects. A study of the positions of the CG over the whole set of flights showed that, in all cases, this was positioned within a maximum distance of 0.15 m from the aircraft's Body Axis Centre. This distance was deemed sufficiently small to simplify calculations and account for any non-measured quantities. *So, for all calculations the Body Axes were assumed to coincide with CG Axes.* To further simplify calculations and avoid inserting additional noise to the data, a 'small perturbation' approach was adopted. Quantities involving products, squared or approximated terms were removed.

$$X_{a\&p} = m(a_x - g_0 \sin\theta) \quad (10-3)$$

$$Z_{a\&p} = m(a_z - g_0 \cos\theta \cos\varphi) \quad (10-4)$$

$$M_{a\&p} = I_{yy}\dot{Q} \quad (10-5)$$

The final set of input variables for the training database is presented in Table 9-2. *Pressure altitude* was used as a measure of the external *Pressure* and *Thrust*<sup>3</sup> was selected to represent *Throttle Position*.

---

<sup>3</sup> The sum of the port and starboard engine thrust components, sourced from the aircraft's on-board Thrust Estimation System

	Variable	Symbol	Units
1.	<i>Angle of Attack</i>	$\alpha$	[deg]
2.	<i>Body Pitch Rate</i>	$Q$	[deg. s <sup>-1</sup> ]
3.	<i>True Airspeed</i>	$V_{TAS}$	[kts]
4.	<i>Vert. Speed Rate</i>	$\dot{W}$	[m. s <sup>-2</sup> ]
5.	<i>Pressure Altitude</i>	$h$	[ft]
6.	<i>Temperature</i>	$T$	[C]
7.	<i>Elevator</i>	$\eta$	[deg]
8.	<i>Thrust</i>	$\tau$	[N]
9.	<i>CG position</i>	$x_{CG}$	[m]

Table 10-2: Input variables for the training database

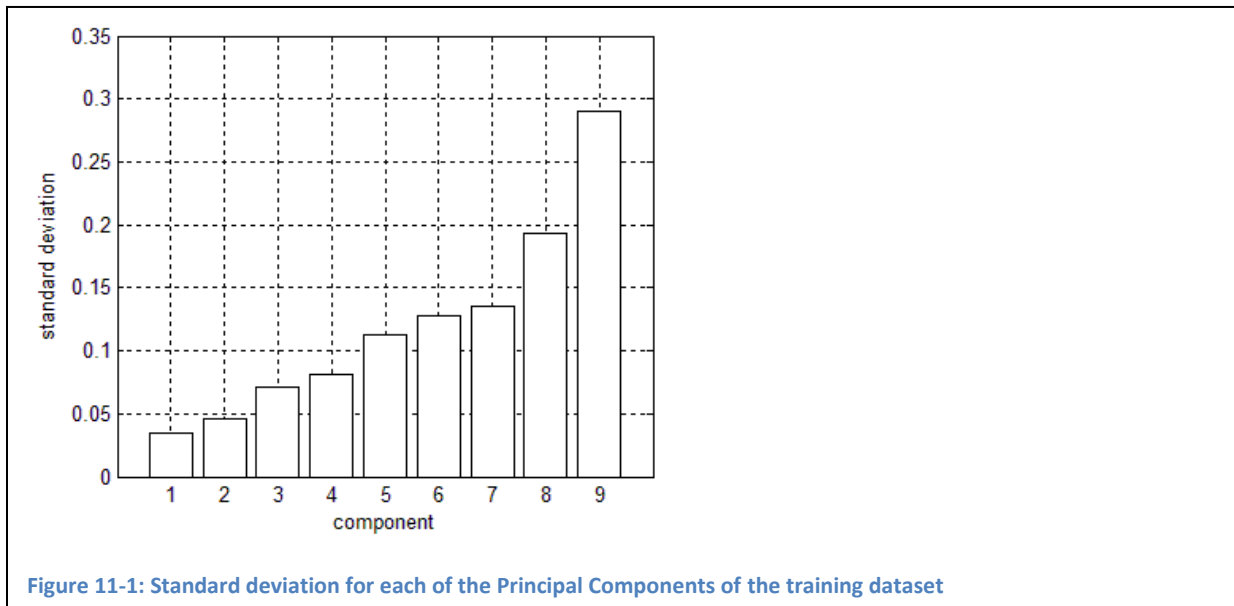
## 11. Modelling of longitudinal forces and moments

Flight data from 55 landings were used to form a training database for the modelling of the longitudinal forces and moments, totaling 62,164 samples. An additional dataset of 5,873 samples was formed using data from 5 flights that were excluded from training to be used for validation. These were picked out of a set of the most recent flights, confirmed to have been realized under mild turbulence conditions (determined via pilot feedback) so as to avoid, to the maximum possible extent, the presence of extreme turbulence in the validation samples. In order to assess the accuracy of the NN predictions as a function of the ‘dissimilarity’ between training and validation flight data, a second validation dataset of 1,939 samples was assembled by 6 flight extracts taken from the 55 ‘training’ landings<sup>4</sup>. This will be referred to as ‘validation dataset B’, as opposed to the term ‘validation dataset A’ which will be used for the first dataset.

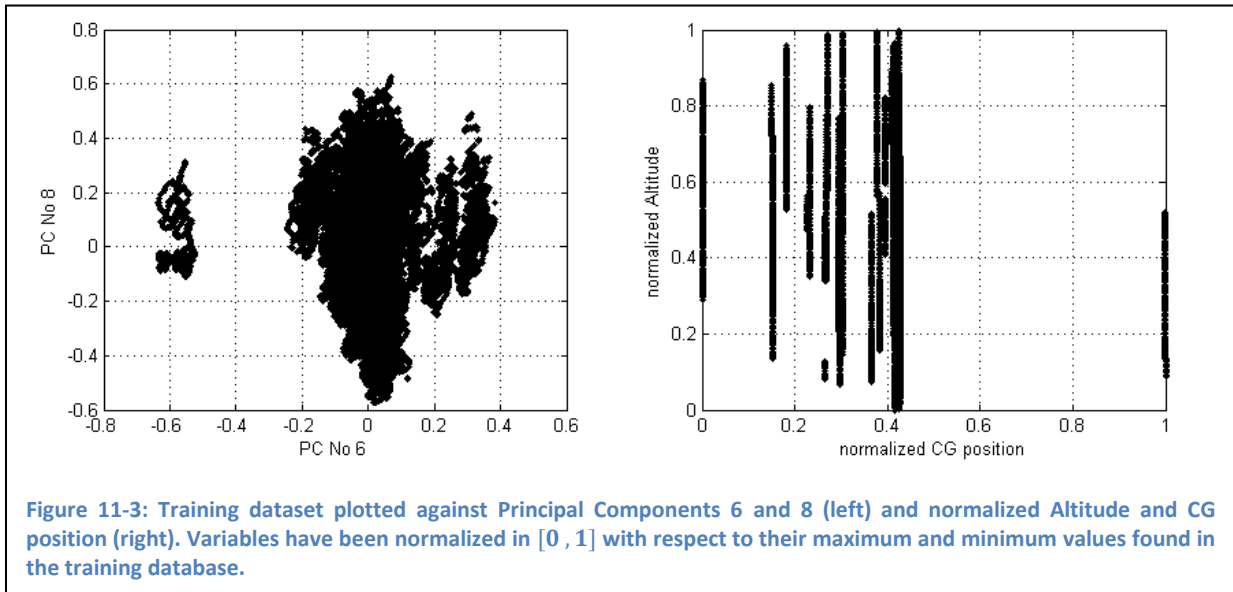
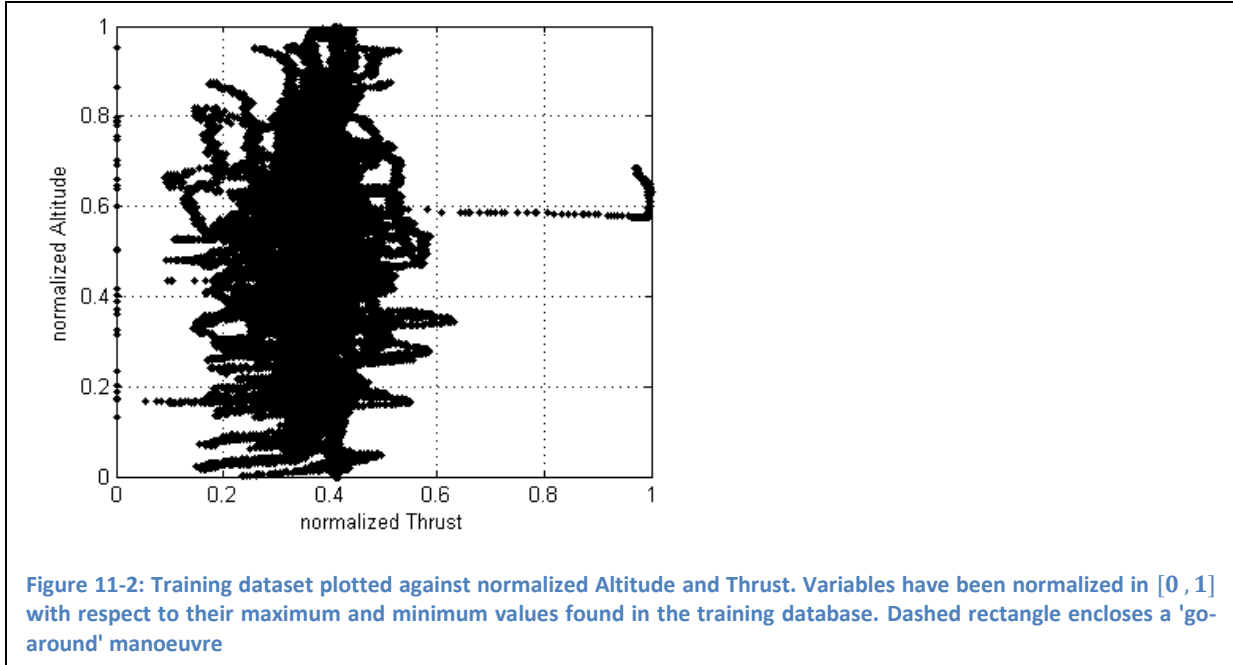
A common set of input variables were selected for the prediction of all longitudinal quantities ( $X_{a\&p}, Z_{a\&p}, M_{a\&p}$ ). To extract information on the distribution of training samples, PCA was applied to the training database. Results, shown in Figure 10-1, indicated that, although a uniform distribution of points could not be reached (for the normalized dataset used, this would occur if the calculated standard deviation  $\sim 0.3$  for *all* PCs), the distribution of variance among the PCs was balanced. Consequently, no ‘strict’ co-relations between the selected input variables were included. Two additional observations were made on the distribution of training samples:

<sup>4</sup> These were also not included in the training database

1. A curve of data points extending out of the main 'core' of the dataset, shown in Figure 10-2, was found to correspond to a 'go-around' maneuver that had been performed in one of the 'training' landings. This gave an indication of the-expected-lack of data points in the high-thrust region.
2. The data plotted against its PCs was found to consist of two clusters (Figure 10-3). The cause of the existence of a second cluster was traced at one of the 'training' landings which appeared to have been conducted at an extreme-aft CG position. Although this should not affect results, all CG positions for the validation flights were checked and found to lie within the main cluster.







Following the results of a parametric study that was conducted in a simulation environment with an existing aircraft model (27), RBFNs with  $m = 150$  neurons were specified for both modelling and error estimation. Validation results for both datasets A and B are summarized in Table 10-1 , Figure 10-4 and Figure 10-5.

Overall, the NN predictions followed closely all variations of forces and moments, while the accuracy achieved was similar in all cases evaluated. Knowing that NN performance is closely linked to the quality of the training data, this signified that the level of noise present in both training and validation databases was low enough to allow for good modelling results.

Combining the above with the fact that INS estimates for  $AoA$  and  $V_{TAS}$  were used, meaning that the main source of noise in the recordings should be attributed to atmospheric turbulence, it may be concluded that the selected model configuration can provide accurate modelling even if the effects of typical turbulence are present in the training data. As expected, RMS error values were lower for Dataset B, since this was, by definition, positioned closer to the training data, however, the degradation in performance between the two datasets was *not* pronounced. Prediction Intervals Correct Percentages (PICPs) in most cases were lower than the specified value of 95%, averaging 79.4% for Dataset A and 85.9% for Dataset B. The cause of this is most probably linked to the distribution of the training samples in the data domain. Although sufficient for modelling, it does not allow for good generalization of results, especially for the more complex function geometries that are associated with noise prediction. Nevertheless, in most cases it was observed that prediction intervals were under-estimated mostly in areas where the discrepancy between flight data and NN predictions was small or negligible.

Quantity	Dataset A		Dataset B	
	RMS Error	PICP [%]	RMS Error	PICP [%]
$X_{a\&p}$ [N]	1313.3	84.7	1232.4	80.9
$Z_{a\&p}$ [N]	2268.9	72.4	1390.2	87.3
$M_{a\&p}$ [Nm]	1063.1	81.0	713.1	89.42

Table 11-1: Validation results for datasets A and B

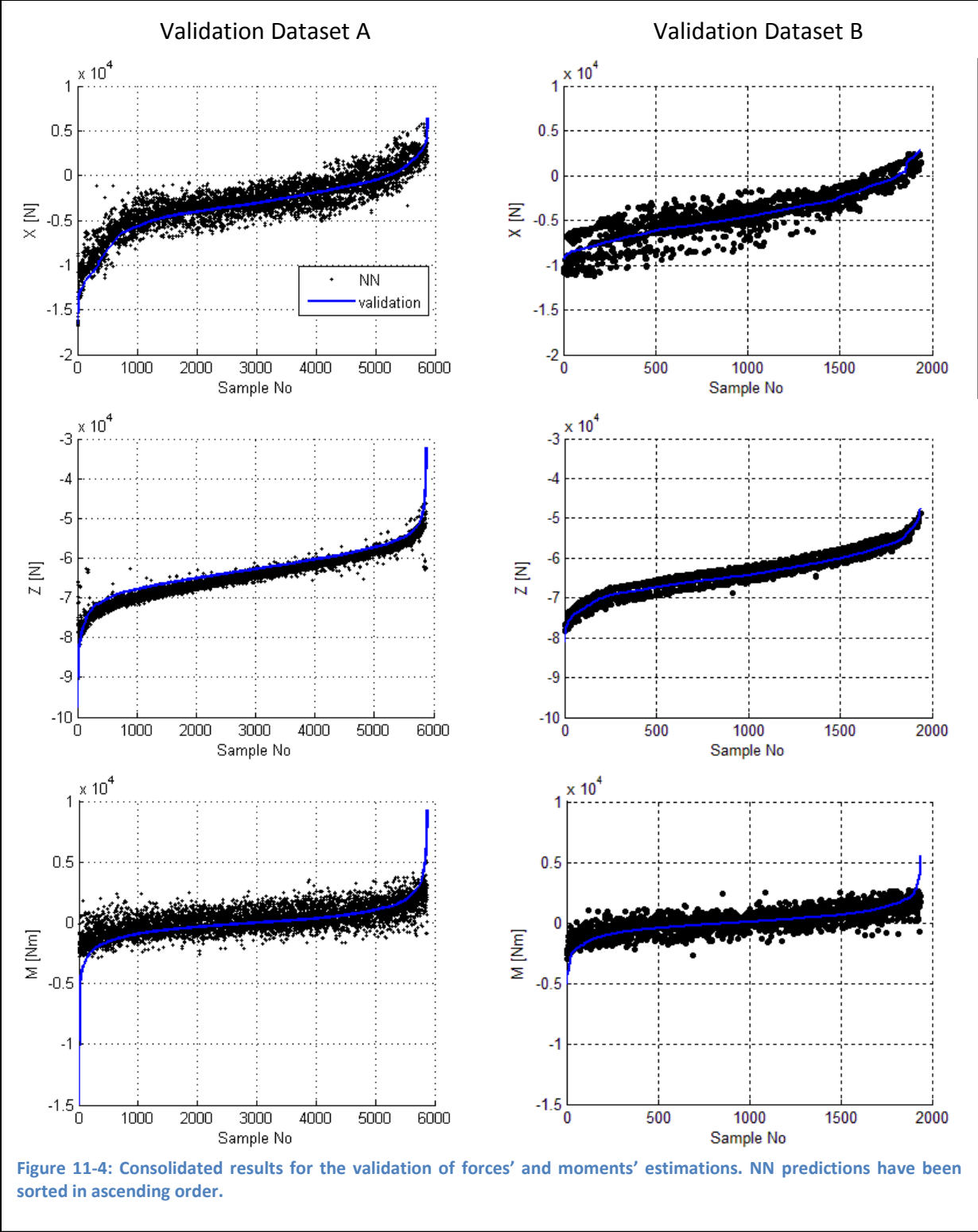


Figure 11-4: Consolidated results for the validation of forces' and moments' estimations. NN predictions have been sorted in ascending order.

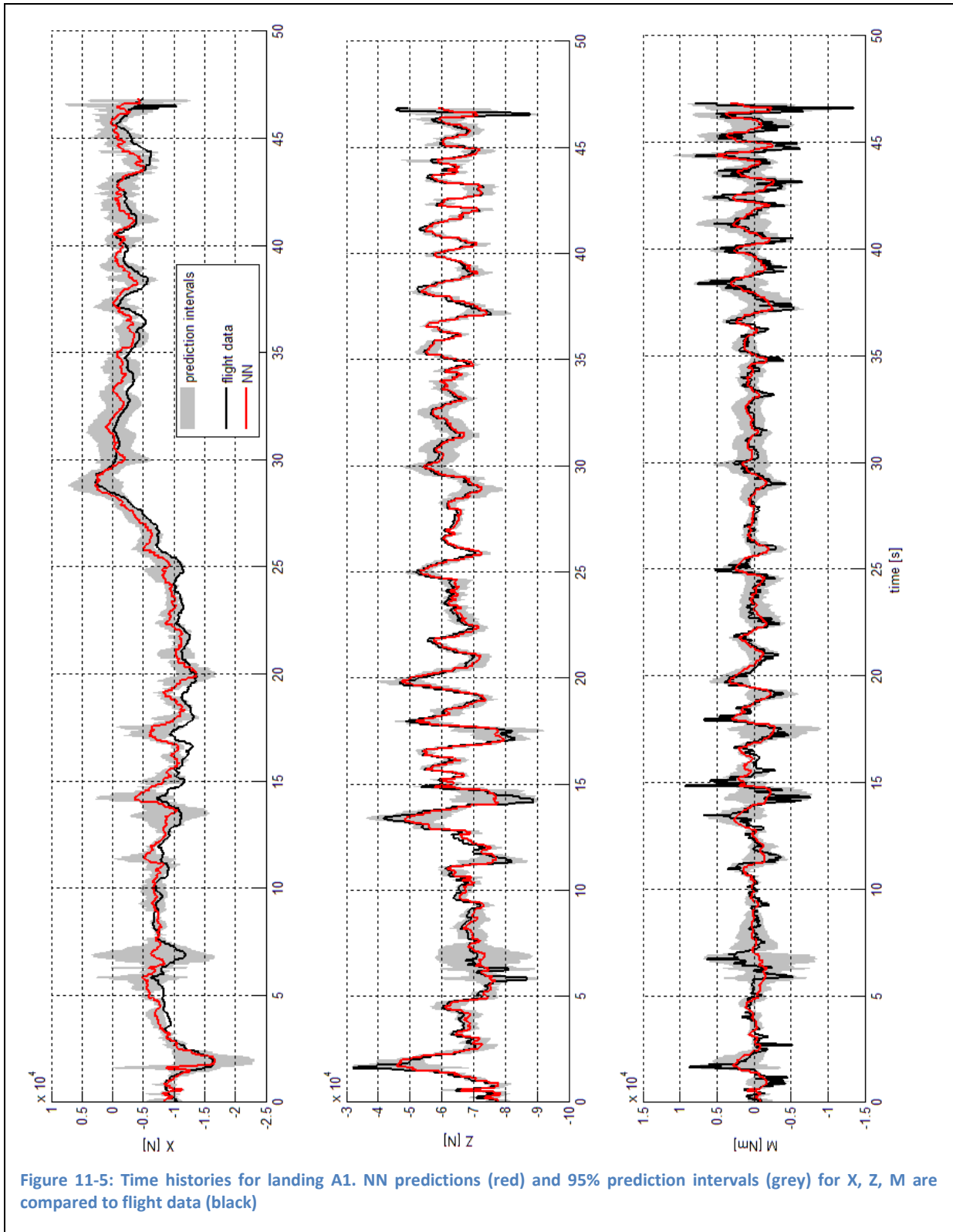


Figure 11-5: Time histories for landing A1. NN predictions (red) and 95% prediction intervals (grey) for X, Z, M are compared to flight data (black)

## 12. Gust Identification

Gust identification involves using the aircraft's on-board sensors as a means to measure the direction and amplitude of the wind components it is subjected to. If air data measurements are available, these can be measured in a direct manner by subtracting the INS estimates for body-axes referenced velocities from the respective values measured by the air data probes. If this is not the case, an alternative strategy can be adopted, which relies on converting aircraft accelerometer readings to wind components. This method, however, pre-supposes that the relationship between the aircraft's response and the relative wind is known. A modelling technique must thus be employed to construct a model of the aircraft's 'inverse dynamics'.

Based on the approach described above, in this study, a set of NNs were used to identify wind gust patterns about the aircraft's longitudinal body axes centre. These were trained to output estimates of  $AoA$ ,  $TAS$  and  $Q$  as a function of the flight condition and the accelerometer readings, converted to Aero-Propulsive Forces and Moments about the aircraft's Body axes. The same training database as for the dynamic modelling case was employed, modified by switching between inputs and outputs. As previously stated Angle of attack ( $AoA$ ), True Airspeed ( $V_{TAS}$ ) and Pitch Rate ( $Q$ ) were set as the NN outputs, while Forces  $X_{a\&p}$ ,  $Z_{a\&p}$  and Pitching Moment  $M_{a\&p}$  were included to the input variable set. Having established a NN model for the aircraft's inverse dynamics, wind gust components were calculated by post-processing the results, using the formulas:

$$WV_w = (\alpha - \alpha_{NN})V_{TAS} \quad (12-1)$$

$$U_w = V_{TAS} - V_{TAS_{NN}} \quad (12-2)$$

$$Q_w = Q - Q_{NN} \quad (12-3)$$

Subscript NN denotes NN predictions. All variables in Equations (12-1) are in SI Units. *Error variances* were also post-processed; using error-propagation:

$$\sigma_{V_w} = V_{TAS} \cdot \sigma_{\alpha} \quad (12-4)$$

$$\sigma_{U_w} = \sigma_{V_{TAS}} \quad (12-5)$$

$$\sigma_{Q_w} = \sigma_Q \quad (12-6)$$

In the absence of a direct reference signal, a time-domain comparison of the produced predictions with a Von Karman turbulence model, configured for 'light' turbulence was performed and is shown in Figure 11-1 and Figure 11-2. In terms of  $U_w$  and  $W_w$ , the maximum observed amplitude was the same for both models at 8 *kts* and 4  $m.s^{-1}$  respectively. A spectral analysis of the signals produced by the two models indicated that the frequency content of both predictions was similar. A comparison between the two models is shown in Figure 11-3 to 11-5. The slightly higher high-frequency content of NN predictions should be attributed to sensor noise, as, in contrast with the turbulence model, NN output relies on inputs from the aircraft's on-board instruments. NN predictions for  $Q_w$  were of significantly smaller amplitude than the respective output of the turbulence model- this should be clearly a matter of the 'shape' of the local atmospheric turbulence. In terms of the examined landings A1-5, the 'mild' turbulence characteristics that had originally been specified were also confirmed by the NN predictions: with only a few exceptions, the amplitude of the estimated wind gust components was low. Gusts could be identified as simultaneous local maxima or minima of the  $U_w, W_w, Q_w$  predictions: it was observed that, probably because of their 'random' direction, gusts tended to affect more than one wind component at a time. An example is given in Figure 11-7. Constant wind components could also be identified (Figure 11-6).

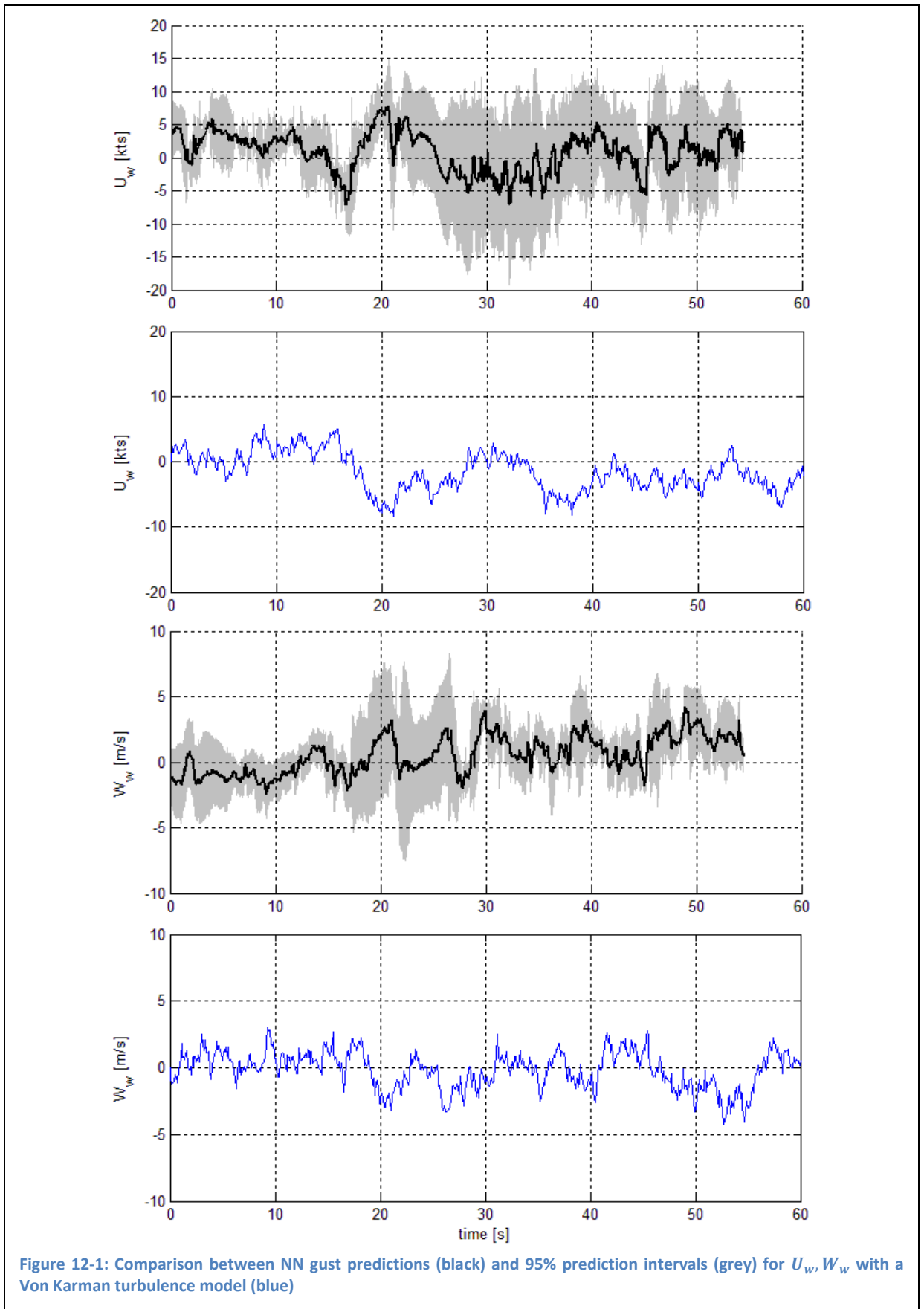


Figure 12-1: Comparison between NN gust predictions (black) and 95% prediction intervals (grey) for  $U_w, W_w$  with a Von Karman turbulence model (blue)

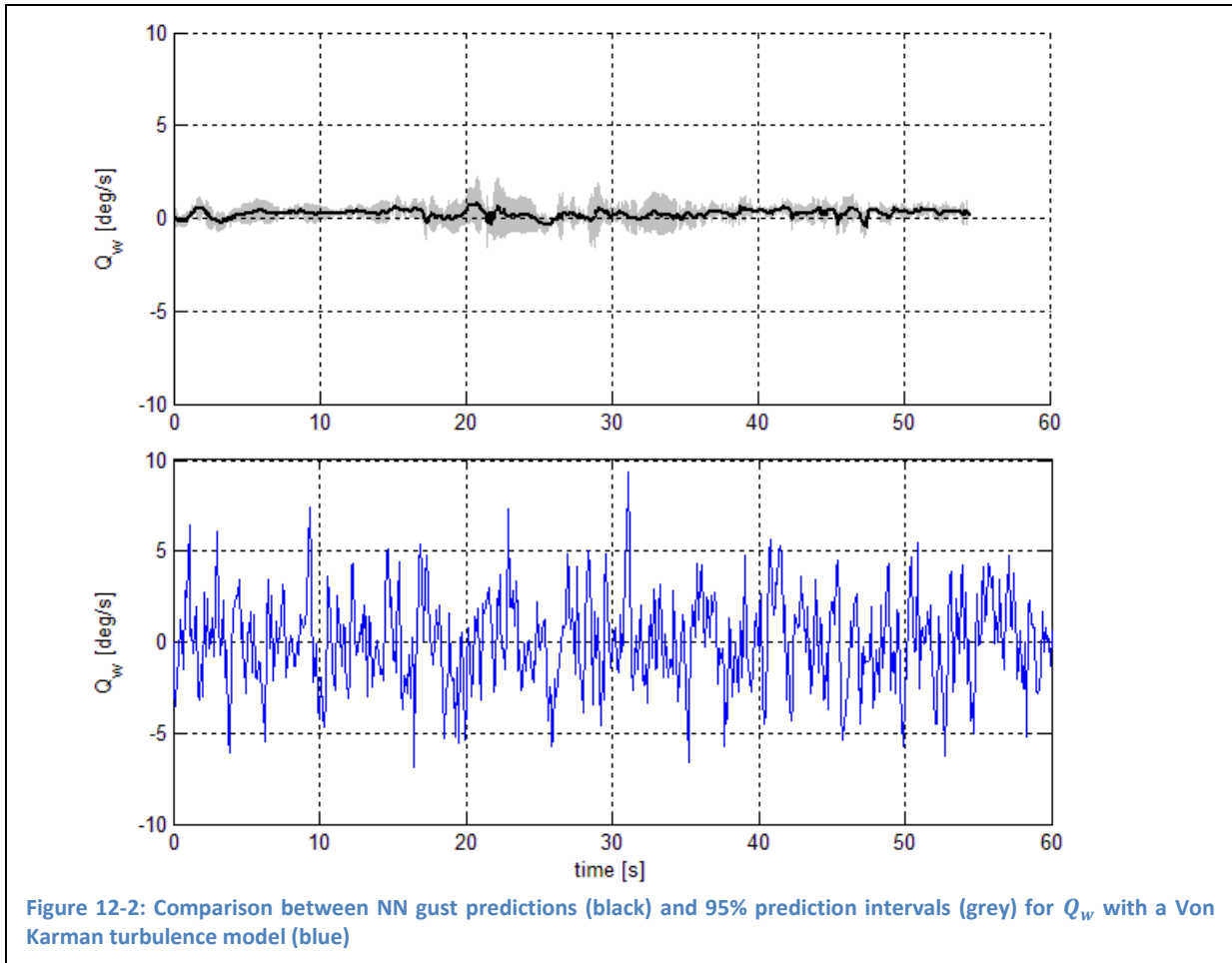


Figure 12-2: Comparison between NN gust predictions (black) and 95% prediction intervals (grey) for  $Q_w$  with a Von Karman turbulence model (blue)



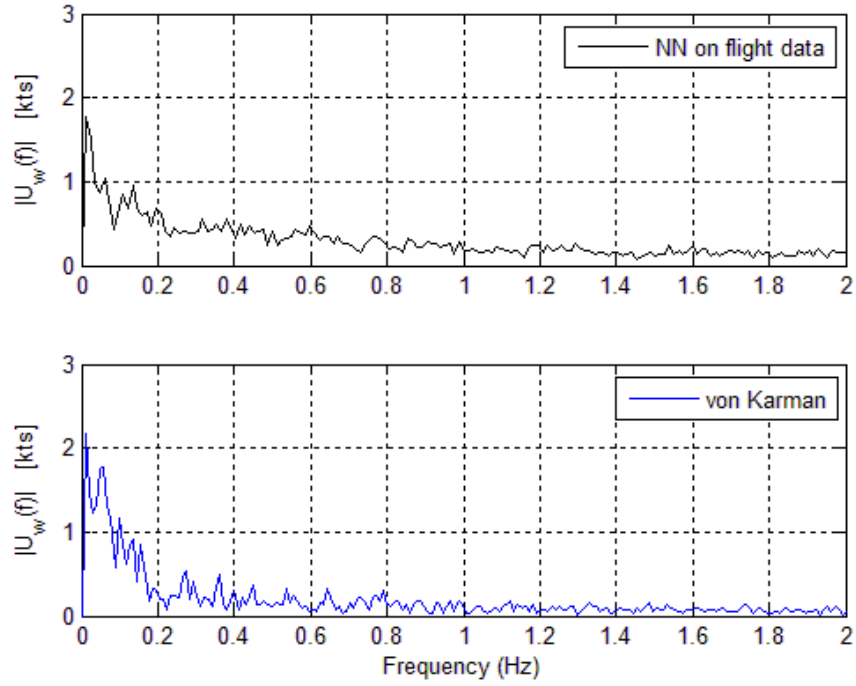


Figure 12-3: Comparison of Single-sided Amplitude Spectrum of  $U_w$  for NN and von Karman predictions

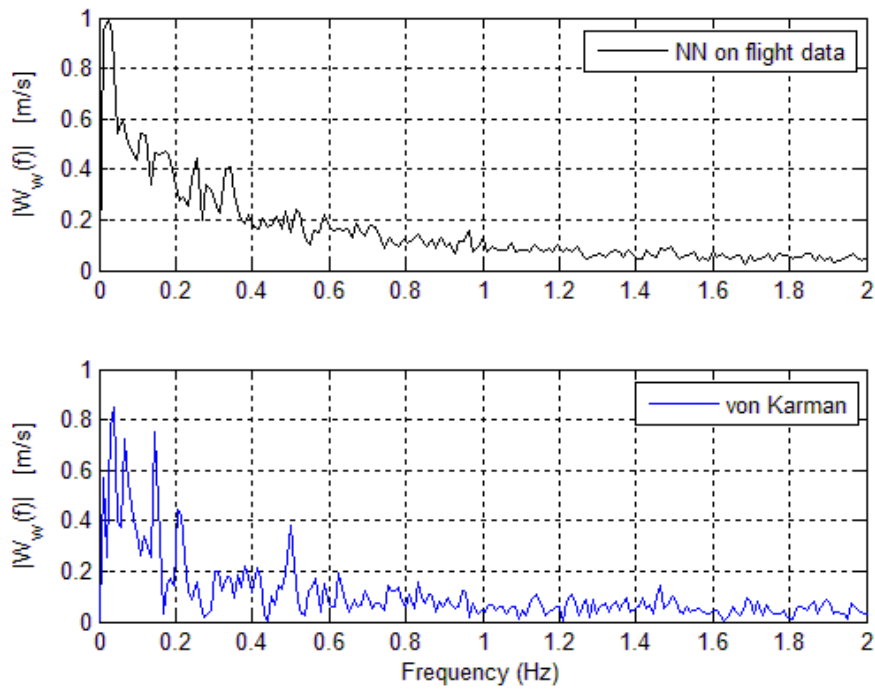


Figure 12-4: Comparison of Single-sided Amplitude Spectrum of  $W_w$  for NN and von Karman predictions

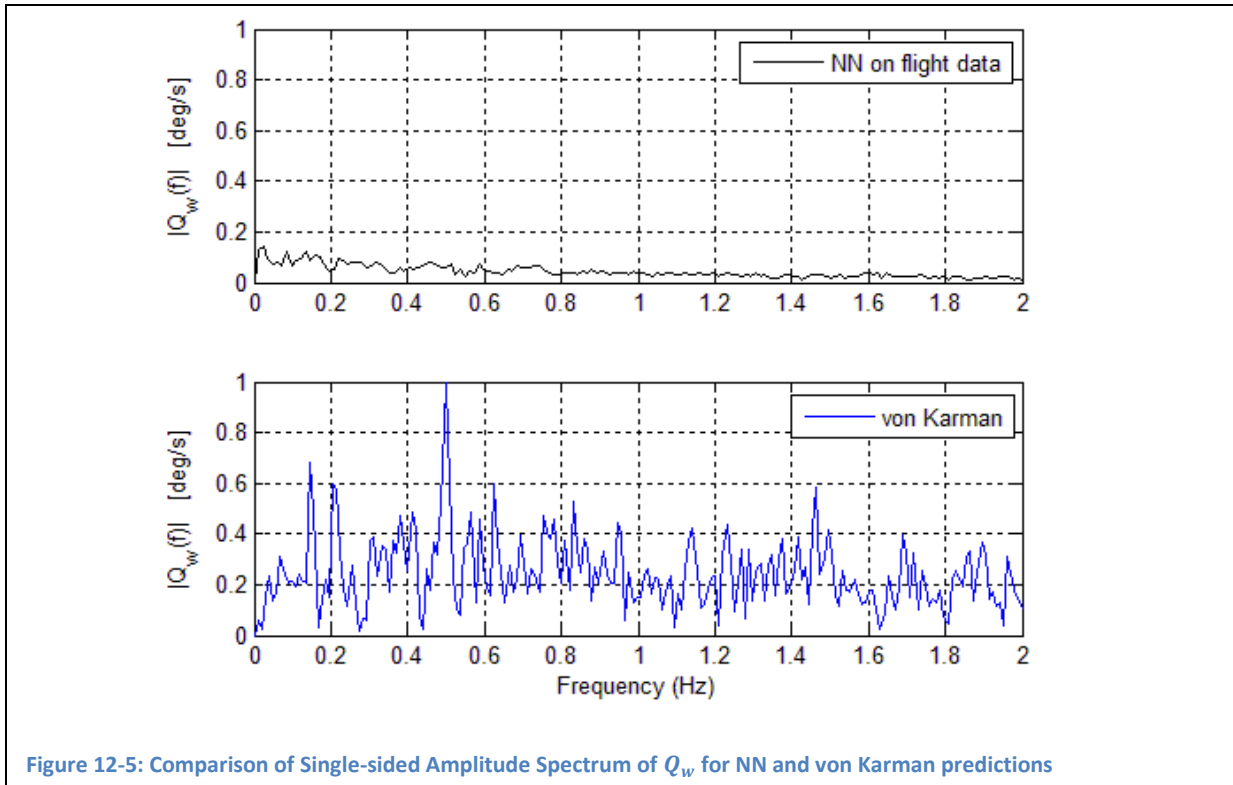


Figure 12-5: Comparison of Single-sided Amplitude Spectrum of  $Q_w$  for NN and von Karman predictions

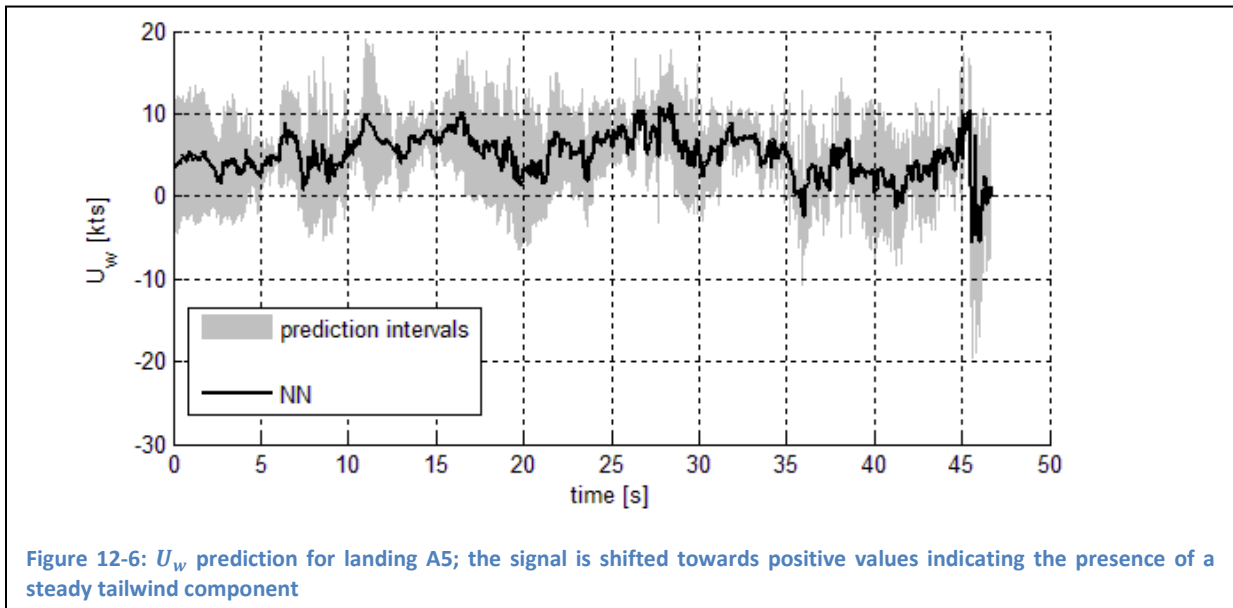


Figure 12-6:  $U_w$  prediction for landing A5; the signal is shifted towards positive values indicating the presence of a steady tailwind component

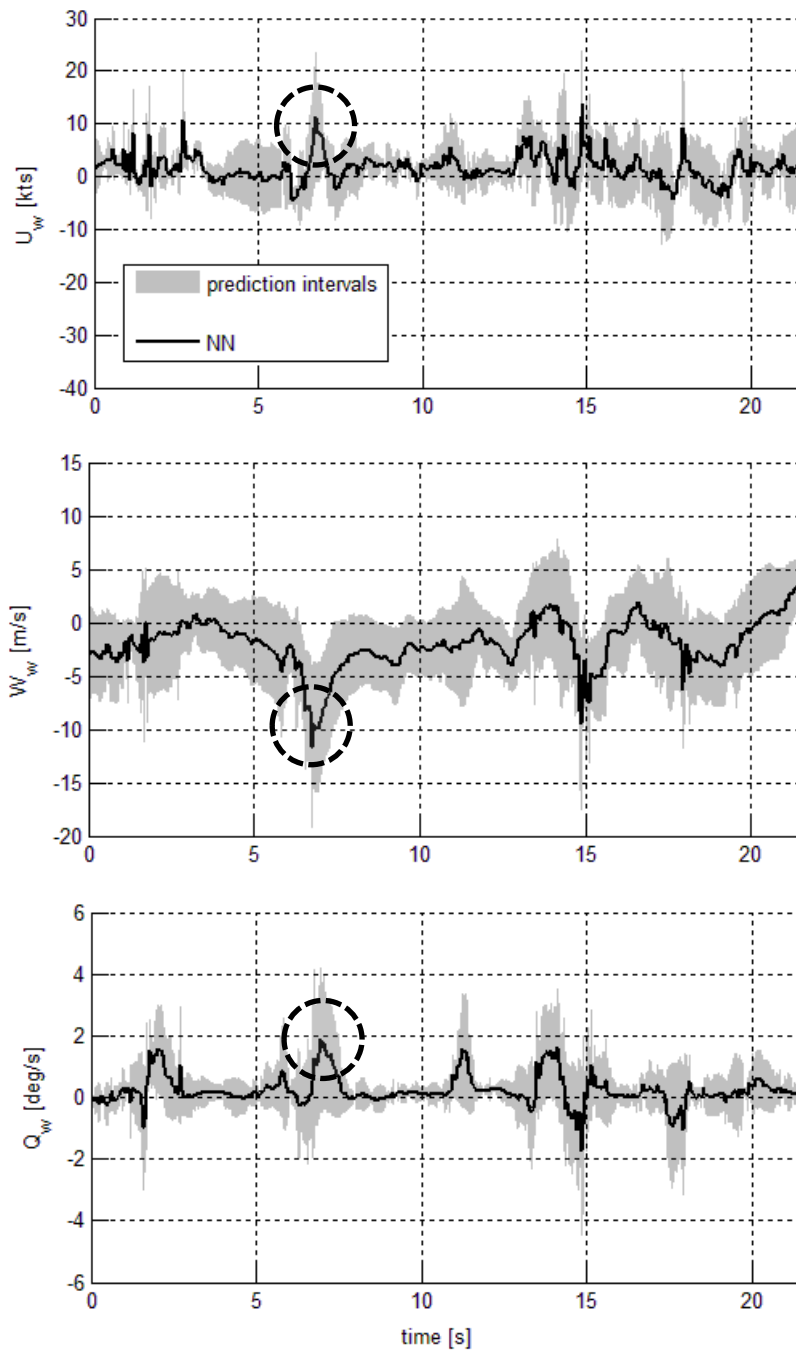


Figure 12-7: Predicted wind components for landing A1. Dashed circles enclose an identified 'gust'

### 13. Conclusions

In this paper a NN-based methodology for dynamic modelling and wind gust identification for a typical aircraft using on flight data recordings was demonstrated. Results indicated that, as intended, the developed NN models produced predictions with an accuracy that would allow

for their use in practical aerospace applications. NNs were verified to be an effective modelling tool which can be successfully applied to aerospace modelling. Provided that adequate testing is conducted to ensure that the accuracy of the produced predictions can be generalized, the inherent non-linearity of NN models offers them an advantage over more traditional modelling techniques, as they require less time and effort to be realized. Models for different applications have been built on the same training database, by simply switching between inputs and outputs, with no other assumptions or conversions.

An analysis of the training database showed that, although expected limitations in terms of the distribution of the observations did exist, overall, maintenance flight data can be effectively exploited to produce accurate NN-based dynamic models of an aircraft.

The efficiency of NN-training can be maximized by effectively combining NN output with that of deterministic models for known quantities. Such a 'hybrid' approach leads to the reduction of the dimensionality of the data domain and, consequently, better results can be obtained from smaller training datasets.

Overall, the developed models were fully practical and usable, although the accuracy achieved is just indicative. Being a function of the quality and quantity of the training data, better results can be achieved if additional effort is dedicated to the proper selection of training samples.

## 14. References

1. Kosmatopoulos E, Polycarpou M, Christodoulou M, Ioannou P. High-Order Neural Network Structures for Identification of Dynamical Systems. IEEE Transactions on Neural Networks. 1995; 6(2): p. 422-431.
2. Shin Y. Neural Network Based Adaptive Control for Nonlinear Dynamic Regimes: Phd Thesis. 2005..
3. Haga R, Matsuura A, Suzuki S, Kono M, Sakaguchi A. Neural Network Based Adaptive Flight Control Using Feedback Error Learning. In AIAA Guidance, Navigation and Control Conference; 2006; Keystone, Colorado.
4. Krüger T, Mößner M, Axmann J, Vörsmann P. Adaptive Flight Control for Unmanned Aircraft Using a Stable Neural Network Observer. In AIAA Infotech@Aerospace; 2010; Atlanta, Georgia.
5. Stastny T, Lykins R, Keshmiri S. Nonlinear Parameter Estimation of Unmanned Aerial Vehicle in Wind Shear Using Artificial Neural Networks. In Guidance, Navigation and Control and Co-located Conferences; August 19-22, 2013; Boston, MA.
6. Puttige V, Anavatti S. Real-time System Identification of Unmanned Aerial Vehicles: A Multi-Network Approach. Journal of Computers. July 2008; 3(7): p. 31-38.
7. Williams-Hayes P. Selected Flight Test Results for Online Learning Neural Network-Based Flight Control System. Edwards, California;; 2004.
8. Nie R, Weiguo Z, Guangwen L, Xiaoxiong L. A Neural Gust Load Alleviator for Aircraft Model Using Active. In IEEE International Conference on Intelligent Computing and Intelligent Systems; 2009; Shanghai.
9. Pashilkar A, Sundararajan N, Saratchadran P. Neuro Controller for Aircraft Auto landing Under Severe Winds and Control Surface Failures. In Aerospace Sciences Meeting and Exhibit; 2005; Reno, Nevada.
10. Nie R, Zhang W, Guangwen L, Xiaoxiong L. Application of Model-following Adaptive Neural Network Control Theory in Gust Load Alleviation. In IEEE International Conference on Automation and Logistics; 2009; Shenyang, China.
11. Dowty Rotor Limited. R.A.C. 5873 Issue II..

12. Powel M. Radial basis functions for multivariable interpolation: A review. In IMA conference on "Algorithms for the Approximation of Functions and Data"; 1985; RMCS Shrivenham.
13. Broomhead DS, Lowe D. Radial Basis Functions, multi-variable functional interpolation and adaptive networks. London;; 1988.
14. Haykin S. Neural Networks: A Comprehensive Foundation Upper Saddle River, N.J.: Prentice Hall; 1999.
15. Hertz J, Krogh A, Palmer R. Introduction to the Theory of Neural Computation Boston: Addison Wesley; 1991.
16. Micchelli C. Interpolation of Scattered Data: Distance Matrices and Conditionally Positive Definite Functions. Constructive Approximation. 1986; 2(1): p. 11-22.
17. Mahalanobis PC. On the generalised distance in statistics. Proceedings of the National Institute of Sciences of India. 1936; 2(1): p. 49-55.
18. Lloyd S. Least Squares Quantization in PCM. IEEE Transactions on Information Theory. 1982; 28(2): p. 129-137.
19. Zaprani A, Livanis E. Prediction intervals for Neural Networks. In 9th WSEAS International Conference in Computers; 2005; Wisconsin, USA.
20. Mazloumi E, Rose G, Currie G, Moridpour S. Prediction intervals to account for uncertainties in neural network predictions: Methodology and application in bus travel time prediction. Engineering Applications of Artificial Intelligence. 2011; 24: p. 534-542.
21. Zio E. A study of the Bootstrap Method for Estimating the Accuracy of Artificial Neural Networks in Predicting Nuclear Transient Processes. IEEE Transactions on Nuclear Science. 2006; 53(3): p. 1460-1478.
22. Carney J, Cunningham P. Confidence and prediction intervals for neural network ensembles. In International Joint Conference on Neural Networks; 1999; Washington, DC.
23. Pearson K. On Lines and Planes of Closest Fit to Systems of Points in Space. Philosophical Magazine. 1901; 2(11): p. 559-572.
24. Jolliffe I. Principal Component Analysis New York, NY: Springer Science+Business Media; 1986.

25. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning: Data Mining, Inference and Prediction New York: Springer Science+Business Media LLC; 2009.
26. Kyriacou S, Asouti V, Giannakoglou K. Efficient PCA-driven EAs and metamodel-assisted EAs, with applications in turbomachinery. Engineering Optimization. 2014; 46(7): p. 895-911.
27. Cooke AK. Simulation model of the NFLC Jetstream 31. Cranfield;; 2006.
28. National Flying Laboratory Centre. Introduction to the Aircraft: G-NFLA Cranfield: Cranfield University; 2012.
29. McCulloch W, Pitts W. A Logical Calculus of Ideas Immanent in Nervous Activity. Bulletin of Mathematical Biophysics 5 (4). 1943;; p. 115-133.
30. Schwenker F, Kestler H, Günther P. Three learning phases for radial-basis-function networks. Neural Networks. 2001; 14: p. 439-458.
31. Myers R, Montgomery D, Anderson-Cook C. Response Surface Methodology: Process and Product Optimization Using Designed Experiments Hoboken, New Jersey: John Wiley & Sons, Inc; 2009.