ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

MANUSCRIPT-BASED THESIS PRESENTED TO
ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY IN ENGINEERING
Ph.D.

BY
NADEAU-BEAULIEU, Michel

FORMULATION OF MATHEMATICAL MODELS USING PARAMETER ESTIMATION TECHNIQUES AND FLIGHT TEST DATA FOR THE BELL 427 HELICOPTER AND THE F/A-18 AIRCRAFT

MONTRÉAL, NOVEMBER 16 2007

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Dr. Ruxandra Botez, director of thesis
Department of Automated Production Engineering at École de technologie supérieure

Dr. Stéphane Hallé, president of jury
Department of Mechanical Engineering at École de technologie supérieure

Dr. Ouassima Akhrif, jury member
Department of Electrical Engineering at École de technologie supérieure

Dr. Karima Bensouda, external jury member
Company Bell Helicopter Textron Canada

THIS THESIS WAS PRESENTED IN FRONT OF JURY AND PUBLIC

ON NOVEMBER 8TH 2007

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOLEDGEMENTS

I would like to express my deep gratitude to Professor Ruxandra Botez who gave me great support through this thesis. During the last four years, she gave me the necessary resources and guidance to progress in this research. I will use the knowledge and wisdom she transmitted to me during the rest of my career.

I had the privilege, during this work, to benefit of the professional assistance of very talented and experimented engineers and researchers from companies which funded the global project: Bell Helicopter Textron and NASA DFRC. For the ground dynamics model detailed in this paper, I had the chance to work closely with Mr. Joey Seto (Bell Helicopter Textron) and Dr. Kenneth Hui (National Research Council), who gave me a great deal of guidance and feedback and Mr. Ed Lambert (Bell Helicopter Textron) for his feedback on the publications of the work on B-427 helicopters. I also wish to thank Mr. Marty Brenner from NASA DRFC for his support in the interpretation of the F/A-18 flight test data.

I also benefited very much from the assistance and collaboration of many other members of the LARCASE team to carry on every aspects of this thesis. Dr. Adrian Hiliuta and Mr. Andrei Popov provided me great support in the initial design of the ground dynamics model and in the final model validation. Other LARCASE team students contributed to the results obtained in this thesis: Mr. Marc-André Cyr, Ms Ruxandra Popescu, Mr. Tarek Merouani, Mr. Abdelatif Khelifi and Mr. Patrick Dionne. The work on the F/A-18 aicraft could not have been carried on without the dedicated collaboration of Miss. Sandrine De-Jésus Mota.

I highly appreciated the two scholarships offered to me ÉTS and by CAE inc. which contributed to pursue this thesis without financial worries.

Finally, last but not the least, I would like to thank my wife for all her love support and encouragements though these four years.

FORMULATION OF MATHEMATICAL MODELS USING PARAMETER ESTIMATION TECHNIQUES AND FLIGHT TEST DATA FOR THE BELL 427 HELICOPTER AND THE F/A-18

NADEAU BEAULIEU, Michel

ABSTRACT

In this thesis, three mathematical models are built from flight test data for different aircraft design applications: a ground dynamics model for the Bell 427 helicopter, a prediction model for the rotor and engine parameters for the same helicopter type and a simulation model for the aeroelastic deflections of the F/A-18.

In the ground dynamics application, the model structure is derived from physics where the normal force between the helicopter and the ground is modelled as a vertical spring and the frictional force is modelled with static and dynamic friction coefficients. The ground dynamics model coefficients are optimized to ensure that the model matches the landing data within the FAA (*Federal Aviation Administration*) tolerance bands for a level D flight simulator.

In the rotor and engine application, rotors torques (main and tail), the engine torque and main rotor speed are estimated using a state-space model. The model inputs are non-linear terms derived from the pilot control inputs and the helicopter states. The model parameters are identified using the subspace method and are further optimised with the Levenberg-Marquardt minimisation algorithm. The model built with the subspace method provides an excellent estimate of the outputs within the FAA tolerance bands.

The F/A-18 aeroelastic state-space model is built from flight test. The research concerning this model is divided in two parts. Firstly, the deflection of a given structural surface on the aircraft following a differential ailerons control input is represented by a Multiple Inputs Single Outputs linear model whose inputs are the ailerons positions and the structural surfaces deflections. Secondly, a single state-space model is used to represent the deflection of the aircraft wings and trailing edge flaps following any control input. In this case the model is made non-linear by multiplying model inputs into higher order terms and using these terms as the inputs of the state-space equations. In both cases, the identification method is the subspace method. Most fit coefficients between the estimated and the measured signals are above 73% and most correlation coefficient are higher than 90%.

FORMULATION D'UN MODELE MATHEMATIQUE PAR DES TECHNIQUES D'ESTIMATION DE PARAMETRES A PARTIR DE DONNEES DE VOL POUR L'HELICOPTERE BELL 427 ET L'AVION F/A-18 SERVANT A LA RECHERCHES EN AEROSERVOELASTICITE

NADEAU BEAULIEU, Michel

RÉSUMÉ

Cette recherche présente différents modèles mathématiques d'aéronefs développés à partir de données d'essais en vol pour trois modèles: la dynamique au sol pour l'atterrissage de l'hélicoptère Bell 427, le comportement des rotors et des moteurs pour le même hélicoptère et la simulation des déflections aéroélastiques de l'avion militaire F/A-18.

La structure du modèle de dynamique au sol du B-427 est déduite par des lois de la physique, dans lesquelles la force normale de contact avec le sol est modélisée par le ressort vertical et la force de friction est modélisée par les coefficients statiques et dynamiques. Les coefficients du modèle sont optimisés afin que sa sortie corresponde aux données d'atterrissage à l'intérieur des marges de tolérance définies par la FAA (*Federal Aviation Administration*) pour un simulateur de vol de niveau D.

Les torques du rotor principal, du rotor de queue et des moteurs ainsi que la vitesse du rotor principal sont estimés par un modèle d'espace d'état. Les entrées non-linéaires du modèle sont construites à partir des commandes du pilote et des états de l'hélicoptère. Les paramètres du modèle sont identifiés par la méthode *subspace* et optimisés par l'algorithme Levenberg-Marquardt. Le modèle donne une excellente estimation des sorties à l'intérieur des marges de tolérances de la FAA.

Le modèle aéroélastique de l'avion F/A-18 est représenté sous forme d'espace d'état, et la recherche concernant ce modèle est divisée en deux parties. Premièrement, la déflection d'une partie de l'avion suite à une entrée d'ailerons différentiels est représentée par un modèle linéaire MISO (*Multiple Inputs Single Outputs*). Les entrées du modèle sont les positions des ailerons et les déflections des autres parties de l'avion. Deuxièmement, un seul modèle d'espace-état non-linéaire est utilisé pour calculer les déflections aéroélastiques des ailes et des volets de bords de fuite. Les non-linéarités sont introduites en multipliant les entrées entre elles avant de les utiliser dans les matrices d'espace-état. Dans les deux cas, les modèles sont identifiés par la méthode *subspace*. D'excellents résultats ont été obtenus où la plupart des coefficients de correspondance et de corrélation sont respectivement au dessus de 73 % et 90%.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABREVIATIONS

ARMA	Autoregressive Moving Average Method
DFRC	Dryden Flight Research Center
ERA	Eigensystem Realisation Algorithm
FAA	Federal Aviation Administration
FCC	Flight Control Computer
MISO	Multiple Input Single Output
MIMO	Multiple Input Multiple Output
NARMAX	Nonlinear Autoregressive Moving Average Exogenous
NASA	National Aeronautic and Space Administration
OBES	On Board Excitation System
OEI	One Engine Inoperative
RFCS	Research Flight Control System
SDV	Singular Value Decomposition

SYMBOLS AND UNITS

A, B, C, D	Matrices describing the discrete state-space model
A	Main rotor disk area (Paper 1) or Parameter proportional to the amplitude of the helicopter oscillations (Paper 1) or Schroeder frequency excitation amplitude (Paper 3 and 4)
AIL_L, AIL_R	Left and right aileron position (Paper 3 and 4)
$Alt_{C.G.}$	Altitude of the helicopter`s center of gravity
Alt_{lowest_pt}	Altitude of the lowest point on the helicopter skids
Cov	Covariance
C_T	Thrust coefficient
C_L	Roll damping
C_M	Pitch damping
C_N	Yaw damping
C_z	Vertical damping
$coll$	Collective position
d_M	Number of component parameters in the model
E	Error in the relationship between G and T
F	Projected noise matrix multiplied by the instrument or Second derivative (Paper 2)
F_K	Kinetic friction coefficient
F_S	Static friction coefficient
F_x	Longitudinal force
F_y	Lateral force
F_z	Vertical force or normal force between the helicopter and the ground
G	Projected output matrix multiplied by the instrument or Gradient (Paper 2)
G_approx	Approximation of G for a system of order n
$G.W.$	Gross weight

H	Impulse response matrix
h	Number of past inputs and outputs used in the prediction
I	Identity matrix
J	Cost function
K	Noise covariance matrix
K_i	Constant, where i is an integer
K_L	Roll stiffness
K_M	Pitch stiffness
K_z	Vertical stiffness
$K_{pitchrate}$	Correction factor on the initial pitching moment related to the pitch rate
$K_{\theta.pitch}$	Correction factor on the initial pitching moment related to the pitch angle
K_z	Vertical stiffness
L	Rolling moment
$LSTB_y$	Left stabilizer lateral position (Paper 3)
lat	Lateral cyclic position
$long$	Longitudinal cyclic position
M	Pitching moment or Mass or Mach Number
N	Yawing moment
p	Roll rate in Body axis system
ped	Pedal position
q	Pitch rate in Body axis system
q_0	Pitch rate at the instant of touchdown
R	Main rotor radius (Paper 1) or Correlation coefficient (Paper 3 and 4)
r	Yaw rate in Body axis system
S	Matrix of singular values
STB_L, STB_R	Left and right stabilizer position (Paper 3 and 4)
s	Number of singular values
T	Projected state matrix multiplied by the instrument or Thrust (Paper 1)

$[T]_{emb}$	Transformation matrix from body axis to modified Earth axis
TEF_L, TEF_R	Left and right trailing edge flap deflections (Paper 3 and 4)
t, Δt	Time and time increment
U	Hankel matrix of past or future inputs or matrix of singular vector
u	System input vector or Longitudinal velocity (Paper 1)
$u^-_{Rel,N}$	Normal velocity of a point relative to the ground before contact with the ground (Paper 1)
$u^+_{Rel,N}$	Normal velocity of a point relative to the ground after contact with the ground (Paper 1)
V	Future noise effect matrix or matrix of singular vector
V_{tan}	Resultant velocity tangential to the ground (Paper 1)
Var	Variance
Vec	Concatenation of the parameters of a matrix into a column vector
$VERT_L$, $VERT_R$	Left and right rudder position (Paper 3 and 4)
v	Perturbation vector on the system outputs or Lateral velocity (Paper 1)
W_i	Weight matrix
$WING_L$, $WING_R$	Left and right wing deflections (Paper 3 and 4)
w	Perturbation vector on the system states or Vertical velocity (Paper 1)
X	Matrix of past or future states
x	System states vector or longitudinal axis or longitudinal position of a point on the helicopter (Paper 1)
Y	Matrix of past or future outputs
y	Output vector or lateral axis or lateral position of a point on the helicopter (Paper 1)
z	Discrete time shift operator or vertical axis or vertical position of a point on the helicopter (Paper 1)
$z_{e,lowest}$	Distance between the lowest point on the helicopter skids and the helicopter center of gravity (modified Earth axis coordinate system) (Paper 1)

$\dot{z}$	Vertical velocity (Paper 1)
$\ddot{z}$	Vertical acceleration (Paper 1)
Δz	Vertical deflection of the skids (Paper 1)
β	Operator defined in equation (2.50) or Exponential decay coefficient (Paper 1)
Δ	Increment in the parameter guess
ε	Restitution coefficient
η	Matrix of past output combinations used in the linear regression
Γ	Observability matrix
θ	Estimated parameter or Pitch angle (Paper 1)
θ_0	Pitch angle at the instant of touchdown (Paper 1)
$\dot{\theta}$	Pitch angle rate of change (Paper 1)
λ	Angle between the longitudinal component and the resultant component of ground tangential velocity (Paper 1) or Levenberg-Marquardt parameter (Paper 2)
ν	Friction coefficient
Δ	Controlability matrix
Π	Projection operator
ρ	Air density
σ	Singular value
φ	Columns of the instrument matrix
Φ	Instrument matrix
ϕ	Roll angle (Paper 1) or phase (Paper 3 and 4)
$\dot{\phi}$	Roll angle rate of change (Paper 1)
Ω	Main rotor rotational velocity
ψ	Yaw angle
$\dot{\psi}$	Yaw angle rate of change

Superscripts

T	Transpose

Subscripts

100%	100 % main rotor speed (not in autorotation)
aero	From aerodynamics
auto	Autorotation
b	Body axis coordinates system
c.g.	Gravity center
damping	From damping
e	Earth axis coordinate system
em	Modified Earth axis coordinates system
eq	At equilibrium (at the end of the landing record)
f	Future value
flight	From aerodynamics, thrust and gravity
friction	From friction
gravity	From gravity
GD	From ground dynamics
i	i^{th} element of the data vector
j	Dummy index (see equation 2.5) or j^{th} iteration (Paper 2)
k	k^{th} timestep prediction of noise effect (line k) on V matrix or k^{th} frequency (paper 3 and 4)
L	Left
Lever_arm	Lever arm between the touchdown point and the center of gravity
Limit	Matrix used to estimate a limit parameter
m	Number of inputs
n	Number of states or order of the system

N	Length of data vector or approximation with a finite amount of data points
o	Number of outputs
oscillations	From the "oscillations" equations
p	Past value
pivot	From the "pivot" equations
R	Right
r	Forward prediction horizon
rel	Relative to the ground
roll	From roll coupling
States	Matrix used to estimate the helicopter states
s	Number of lines of the instrument matrix
skid	Helicopter skid
skid, fwd	Forward part of the skid
skid, aft	Aft part of the skid
Thrust	From thrust
threshold	Point where the pitch stiffness increases
wt	G matrix with weight functions added
x	Longitudinal direction
y	Lateral direction
z	Vertical direction

Symbols

∂	Partial derivative operator
$\perp$	Perpendicular projection
∞	Infinity
$+$	After a contact

-	Before a contact
$\|x$	Output dependant on the parameter x

INTRODUCTION

In the aerospace field, is often very important to generate mathematical models to represent different dynamic phenomena. Such a mathematical model can be used in many applications such as a flight simulator or a flight control system. In this thesis, four mathematical models are built from flight test data to solve specific problems related to aircraft design and operation.

Two approaches are used to generate a mathematical model: analytical and system identification. The first model developed in this thesis uses a semi-analytical approach where the model structure has been derived from first principles, but its coefficients are optimised to match the flight test data. The second to fourth model structures and coefficients were derived by the system identification approach.

The first model is a ground dynamics model for the Bell-427 helicopter in which the forces and moments after touchdown are calculated for different landing cases. The second model simulates the responses of some parameters related to the main rotor, tail rotor and engine of the helicopter B-427 following control inputs from the pilot. The third model computes the deflections of different structural surfaces of a modified F/A-18 aircraft using a multiple inputs single output model (MISO) where the model inputs are the differential aileron deflections and the known deflections of aircraft structural surfaces. The fourth model is an improvement of the third model where all the structural surface deflections of the F/A-18 are computed simultaneously in a multiple inputs multiple outputs MIMO model.

In the analytical approach, a mathematical model is built from basic physics laws such as Newton laws of motion, thermodynamics, etc. The main advantage of this approach is that each aspect of the mathematical models is understood and related to the theory. The validity of mathematical models derived from analysis is also well known and depends

on the initial assumptions. Once these models are built, they provide a reasonable solution for a wide range of conditions and, since each parameter in the model was derived using physics laws, the model can be easily updated if the method to estimate a given parameter is improved. The main drawback of using only an analytical model is that it often represents an over-simplification of the true system dynamics. The first cause of this over-simplification is that most of the equations from theory were derived using assumptions which may have significant effect on the results. The other reason is that most physical systems are extremely complex and have many components. The characteristics and interactions between these components is not always known.

The system identification method consists of building a mathematical model directly from experimental data. In this case, a model structure is defined and its parameters are selected to ensure that the resulting model output matches the flight test data output for specific inputs. The main advantage of system identification over analytical methods is that since the mathematical model is derived from flight test data, it will capture any complex features of the system which would not have been found otherwise. The standard procedure used in any system identification processes is illustrated in figure 1.1:

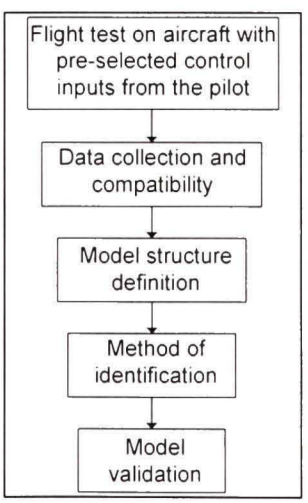

Figure 1.1 ***Basic procedure to follow in the system identification process***

As illustrated in figure 1.1, the first step of system identification consists in deciding which control input should be used to generate an appropriate system response. The choice of appropriate control input mostly depends on the intended model application, which was pointed out by Jategaonkar (2006). These control inputs should generate a response which contains as much information as possible about the system behaviour within the intended application. These control inputs are applied on the aircraft during the flight tests where the aircraft response is measured. The test engineer should be sure of the data excellent quality and should know very well the sensors limitations and the measurements errors corrections dependent on their locations. An extensive description of these corrections was given by Crisan (2005). The user must verify the data compatibility by a process called *flight path reconstruction* described by Jategaonkar (2006) where accelerations measurements are integrated and compared to speeds measurements.

The next methodology step is the definition of a model structure. According to Jategaonkar (2006), the models found by use of system identification methods are classified in three categories: *white box* models, *black box* models and *grey box* models. With the white box model, the identified model is the closest to a pure analytical model. In this case, an analytical set of equations describing the system is derived from first principles, where some physical parameters are left unknown. These physical parameters are identified using flight test data and the parameters values which match the data can later be compared to these parameters estimated values found from analysis or other sources such as wind tunnel testing.

At the other extreme, the *black box* model structures are identified by using input-outputs matching algorithms without a priori information on the true system dynamics. In these types of models, the structure can be a state-space model of arbitrary order or a neural networks model. Another possible structure is the *grey box* model. In this case, the large scale behaviour of a physical system may be assumed, but there is no

knowledge on the interactions of its sub-systems. In this case, the global model structure may be derived analytically and the subsystems may be modelled with *black box* models. It is typically much easier to design a black box model than a white box model due to the fact that there are systematic procedures to conceive a black box model by use of only input and output data, while a white box model requires extensive analytical efforts. On the other hand, the validation of a black box model is limited to the number of flight conditions used to build it, whereas a white box model can be used with more confidence outside these flight conditions if one knows the variation of its parameters, and the type of assumptions to be made.

Once the model structure is determined, the next step is the parameters values identification. If the model structure is a white box model, this identification can be done by using priory values of the parameters and iterating by use of a minimization algorithm until the error between the model output and the flight test data output is minimum. This method is called an *Output Error Method* and is described by Jategaonkar (2006). If the model is a black box, the initial values of the parameters may either be random, as in neural networks algorithms, see Howard et al. (2006) or may be estimated by use of the *subspace method* detailed by Ljung (1999) if the model structure is under state space form.

The last step of the system identification process is the model validation. The best method to validate a model is to simulate it using another data set. The control inputs from another data set are inputs to the model and the model response is compared to the data set. As stated by Jategaonkar (2006), this validation method is called the *acid test*. In some cases, a model may match very well with the data used to build it, but may fail the *acid test*. This failure may indicate that the model is over-parameterised, which means that fictitious parameters are attempting to model the system noise during the identification process. A solution to this problem is to simplify the model until it gives very good results with flight test data used in its identification, as well as with other

separate data set for its validation. It is not always possible to set data aside for the *acid test* due to the fact that the data quantity is limited. For this reason, is a good practice to make a model as parsimonious as possible, and its parsimony can be tested with methods which were initially developed by Akaike (1969), and very well explained by Ljung (1999).

All journal papers presented in this thesis use the above described identification steps to obtain a good mathematical model from flight test data. In the first paper presented in this thesis, a ground dynamics model formulation is developed for the Bell-427 helicopter simulator from landing tests data. The Bell-427 is a two engines four blades commercial helicopter which was first delivered in January 2000. The ground dynamics model was developed and implemented in a level D full flight model developed in collaboration with Bell Helicopter Textron and the Canadian National Research Council. It was validated for one engine inoperative landings (OEI) and autorotation landing. The ground dynamics model was identified with a *grey box* model, in which the overall system behaviours were derived from physics.

The second journal paper presented in this thesis was also realized by use of the Bell-427 flight data, but, this time, the parameters related to the rotors and engine were modelled. These parameters are the main rotor torque, tail rotor torque, engine torque and main rotor speed. The model structure is a state space model of arbitrary order, and is called a *black box* model. Non-linearities are introduced in the model by combining the inputs as higher order terms and using these terms as the inputs of the state-space equations. The model parameters were identified by use of the subspace system identification method implemented in MATLAB® by Ljung (1999). This method is a non-iterative algorithm which finds the system parameters from the input and output vectors by reconstructing an observability matrix. The performances of the subspace identification method alone (without optimization) with respect to the subspace method followed by an optimization are evaluated. The resulting mathematical model was

implemented as a simulation, for a flight simulator application and as a prediction for a carefree manoeuvring flight control system application.

The third paper of this thesis illustrates how a state space model whose parameters are identified by use of the subspace method can be used to simulate the structural deformations of different parts of a F/A-18 aircraft. This aircraft was modified, as described by NASA Dryden Fight Research Center website (2006) with additional actuators on the leading edge flap to split it between inboard and outboard sections. The wing was modified with thinner wing panels to render the wing more flexible and to allow its tip to twist up 5^0. At high dynamic pressures, the F/A-18 control surfaces were used as tabs that are deflected into the air stream to produce favourable wing twists. This technology was able to use the air stream energy to twist the wings and thus to minimize the control surfaces motions.

In this thesis, a mathematical model is built to identify the wing deflections and the aircraft control surfaces following differential ailerons control inputs. Different Multiple Inputs Single output (MISO) models are generated for the different aircraft structural surfaces. For each of the aircraft structural surfaces, the inputs are the differential ailerons and the deflections of the other aircraft structure surfaces. The fourth paper represents an improvement of the third paper where the deflections of aircraft structural surfaces are represented with one single Multiple Inputs Multiple Outputs MIMO mathematical model.

This thesis is organized as follows: In Chapter 1, a literature review is presented for each application journal paper exposed in this thesis. In Chapter 2, the theory applied for each journal publication is summarized. The emphasis of Chapter 2 is the *black-box* system identification method extensive explanation which is used in the second, third and fourth papers. The first paper on the ground dynamics model is introduced in Chapter 3 and is presented in Chapter 4. An introduction to the second paper is given in Chapter 5, and

the second paper on the identification of the helicopter rotor related parameters is presented in Chapter 6. The first paper on the identification of the F/A-18 structural deflections following a control input is introduced in Chapter 7 and is presented in Chapter 8. The second paper on the F/A-18 is introduced in Chapter 9 and is presented in Chapter 10. Finally, conclusions are presented and followed by future work recommendations.

CHAPTER 1

LITERATURE REVIEW

This literature review is organized as follows: in the first section, the past research on the ground dynamics of helicopters is presented followed in the second section by the past research on the modelling and prediction of the main rotor, tail rotor and engine related parameters. The third section presents the past research on the identification of aircraft aeroelastic models. The last section is a brief description of the originality and impact of this thesis with respect to the previous research.

1.1 Ground dynamics helicopters model

Two types of bibliographical research reviews are presented in Sections 1.1.1 and 1.1.2. The first section reviews theoretical formulations of rigid body collisions and contacts which can be applied to an helicopter impacting the ground. The second section reviews the applications of these theories to the ground dynamics of a helicopter in the past.

1.1.1 Past research on rigid body collision

There are only a limited number of publications in the field of helicopter ground dynamics. For this reason, it is necessary to refer to rigid body collision and contact theories in order to find theoretical formulations of an impact between an object and the ground. Three existing methods can be used in touchdown modeling and in the contact forces calculations between the helicopter and the ground, and they are: the *Impulse-constraint method*, the *Impulse method* and the *Penalty method*.

The *impulse-constraint method* was developed first by Baraff (1989, 1993), and uses an instantaneous impulse to model the impact between a rigid body and the ground. The magnitude of this impulse depends on the restitution coefficient values. Following

touchdown, when the body stays on the ground, the tangential constraint forces are solved analytically which requires a high number of iterations.

The *impulse method* is a variant of the impulse-constraint method first developed by Mirtish (1996). With this method, when the object is in contact with the ground, no constraint forces are applied and the object motion is stopped by use of multiple impulses. This method is easier to implement as it does not require iterations, but it is less physically exact than the impulse-constraint method. The impulse method is computationally expensive when the object has more than one contact point with the ground and, therefore, is subjected to a high number of impulses. The method was further improved by Guendelman al. (2003) by use of lateral impulses to calculate the static and dynamic friction coefficients.

The third method, called the *penalty method*, as first implemented by Moore and Wilhelms (1988) consists of modeling the ground as a spring pushing the object upward as it penetrates into the ground. This method is less physically exact than the impulse method, but unlike the impulse method, it does not need to evaluate the exact collision time between an object and the ground.

1.1.2 Past research on helicopter ground dynamics

The penalty method was chosen by Johnson (1997) to model the ground dynamics of an unmanned helicopter because the small ground penetration present in the penalty method could be interpreted in this case as skid deformation.

In Johnson's model, when the helicopter was on the ground, a dynamic friction force was considered when the total force applied on any of the lateral springs exceeded the maximum static friction force or when the lateral velocity of any of these springs was non-zero.

Johnson did not specify how the stiffness and damping of each spring was selected. This unmanned helicopter model was not validated by use of landing test data.

In this thesis, a ground dynamics model was developed and validated based on landing data of a real helicopter. Use of this new approach has never been considered until now.

1.2 Identification and prediction of parameters related to the main rotor, tail rotor and engine

A great deal of modeling effort has been made in the past decade to model the responses of main rotor torque, main rotor speed and engine torque of helicopters, especially to design envelope protection design control systems. Envelope protection control systems ensure the remaining of the rotorcraft parameters within their prescribed design limits. During helicopters operations, many parameters must be restricted between their minimum and maximum limits. In this thesis, these parameters might be the main rotor torque, main rotor speed or engine torque. If the operational limit of one parameter is exceeded, it can have a detrimental effect on the helicopter fatigue life, its handling qualities and safety. In order to prevent such a limit violation, it is necessary to know the relationships between the future control inputs and the future value of the limit parameters. Inverting this model provides the future control inputs which would result in a limit violation. In 1995, Howitt used a simplified mathematical model to estimate the engine torque and main rotor torque of the BO-105 helicopter following a collective step input, and designed carefree control laws based on this simplified model. Much of the research efforts that followed used Neural Networks to predict limit parameter future values. The authors who made most of the research in this aspect are Menon et al. (1996), McCool (1998), Horn and Prasad (1999), Yavrucuk et al. (2001 and 2002) and Horn and Sahani (2002, 2004, 2005). More details on each author contribution is available in chapter 6.

As described in the previous literature review, most of the previous work involving torque prediction concerned the use of Neural Networks to identify a proper model. In this thesis, it is attempted to simulate and predict the future values of the helicopter parameters by use of a non-linear state-space model identified with the subspace system identification method. The subspace method is a relatively recent system identification method that has the capability to identify the parameters of a system without requiring any iterations. It is therefore a very efficient method which does not require a-priori estimates of the system parameters. A detailed description and literature review of the subspace method is presented in Chapter 2.

1.3 Identification of the structural deflections on an F/A-18 aircraft

The active aeroelastic wing research project requires the use of system identification methods to identify the structural deformations of different structural part of an F/A-18 aircraft following a control input given by the pilot. There has been a growing interest in the past decade in the identification of aeroelastic deformations of aircraft from flight test data.

The Autoregressive Moving Average Method (ARMA) and the Neural Networks theories were used by Sung et al. (2005) to identify the flutter behaviour of a transonic wing. The flutter dynamics of a pitch-plunge system subjected to limit cycle oscillations was modeled by Kukreja and Brenner (2006) with non-linear models. The non-linear models used were the Nonlinear Autoregressive Moving Average Exogenous (NAMAX) models. The dynamics of a flexible wing model was identified by Silva et al. (2006) using the impulse response method. The Eigensystem Realisation Algorithm (ERA) followed by an output-error minimisation method was performed based on a large flexible aircraft by Le Garrec et al. (2001).

The subspace method was used by Brenner (1997) to identify the controls' effects on the rigid modes of the F/A-18 aircraft from flight flutter tests. In this method, the ailerons

were excited by use of Schroeder frequency sweeps, and the accelerations were filtered using a wavelet transform, thus the aircraft roll response was identified in both time and frequency domain.

The subspace identification method is used to identify the F/A-18 aircraft surfaces structural positions from flight flutter tests following to the differential ailerons Schroeder excitations. The focus of this research is on the identification of the flexible mode oscillations of the F/A-18 aircraft rather than on its rigid modes, such as the roll rate, studied by Brenner (1997).

1.4 Innovation and impact of this thesis

The research presented in this thesis represents an improvement over previous research in many respect. The first paper, on ground dynamics, presents a new formulation based on the penalty method used by Johnson (1997) in an autonomous helicopter. As it was necessary to qualify the model for a level D flight simulator, it was necessary to derive a much more elaborated model to match the FAA tolerance bands for many different landing cases. Another challenge involved with this model was its integration with a global flight model based on stability and control derivative to avoid instabilities in the model output during the transition between the flight model above the ground and the ground dynamics model.

In the second paper, as it was shown in the literature review, black box methods such as Neural Networks were widely used to predict the future value limit parameters on different rotorcrafts, but it was never done with a state-space model identified with the subspace method. The subspace method has some very attractive advantages over the Neural Networks method. The main advantages of the subspace method over Neural Networks is the fact that it is non-iterative method and that it does not requires a-priori estimate of the model parameters. This implies that in an identification process using the

subspace method, the model will never converge to a local minimum as it could happen in an iterative method with poorly chosen initial parameter guess or minimisation algorithm.

In the third and fourth papers, a State-Space model using the subspace method was used to identify the aeroelastic deflections of different surfaces on an F/A-18 following an excitation on its control surfaces. The application of identification methods to solve aeroelastic problems is very recent in the literature. In most cases, identification methods have been used to model simplified sub-systems such as a pitch-plunge system or a flexible wing. This thesis presents the first research that was done on the structural deflection of every surfaces on the F/A-18. The results from this research should contribute to the development active control systems which could control the structural deflections of an aircraft to improve its flying qualities, structural resistance and safety.

CHAPTER 2

BACKGROUND THEORY

This section will explain thoroughly theoretical concepts which are necessary for the explanation of the following chapters.

2.1 Ground dynamics model theory

The ground dynamics model structure developed in this paper is based on the penalty method as used by Johnson (1997), who modelled the impact between a model helicopter and the ground with a vertical spring, as shown in the following equation:

$$F_{z,GD} = -C_z \dot{z} - K_z \Delta z \qquad (2.1)$$

where $F_{z,GD}$ is the vertical force between the ground and the helicopter from the ground dynamics after the touchdown, Δz is the helicopter skids deformation, $\dot{z}$ is the vertical velocity of the helicopter after the touchdown, K_z is the vertical stiffness and C_z is the vertical damping of the helicopter. In equation (2.1), the coordinate z is defined positive when is downward oriented. From equation (2.1), when the helicopter touches the ground, the skids deflection Δz is positive and, since the helicopter center of gravity has a downward velocity, the vertical velocity $\dot{z}$ is positive. Since the stiffness and damping coefficients are always positive, equation (2.1) gives a negative vertical force pointing upwards. The equation describing the vertical force exerted on a helicopter at touchdown is therefore the same as the force exerted by a spring and damper system. When the helicopter touches the ground, it is subjected to three forces and three moments which arise from the interactions between the ground and the helicopter. In the ground dynamics model, expressions for forces and moments were derived from basic principles given by equation (2.1).

Each of these derived equations are written as function of coefficients needed to be determined and will be discussed in the ground dynamics model theory. In equation (2.1), these coefficients are the system stiffness coefficient K_z and the system damping coefficient C_z, and are identified by minimizing the error between the simulated response of the helicopter and its measured response. The ground dynamics model can be described as a *white box* model, see Jategaonkar (2006). In some cases, in order to obtain a model which matches the landing data, it was required to formulate some of these coefficients as function of other parameters without necessarily using physical principles, the sub-models expressing these coefficients could therefore be thought as *black box* sub-models. The overall ground dynamics model can therefore be described as a *grey box* model because its overall structure was determined from physical principles and some of its coefficients were functions of parameters which were not calculated directly from physics.

The greatest challenge encountered in the ground dynamics model formulation was the formulation of an appropriate model structure. The coefficients values were further iterated to ensure that the model matched the landing data.

2.2 Theory related to the subspace identification algorithm used in the last three papers

Both the second paper on the limit parameters identification on the B-427 helicopter and the third and fourth papers on the structural deflections identification on the F/A-18 are black box *state-space* models. The parameters describing these models were found by use of the subspace system identification method. This section will explain in details the theory behind the subspace method, which is an efficient non-iterative algorithm which determines the model parameters directly from inputs and outputs data.

Before describing the subspace method, it is necessary to define the general formulation of a state-space model.

2.2.1 State-Space model definition

In general, a discrete state-space model can be expressed in the following equations:

$$x(t+\Delta t)_{n\times 1} = A_{n\times n}x(t)_{n\times 1} + B_{n\times m}u(t)_{m\times 1} + w(t)_{n\times 1} \tag{2.2}$$

$$y(t)_{o\times 1} = C_{o\times n}x(t)_{n\times 1} + D_{o\times m}u(t)_{m\times 1} + v(t)_{o\times 1} \tag{2.3}$$

Where t is the time, Δt is the sample time of the simulation, $u(t)$ represents the system inputs where m is the number of inputs and $y(t)$ represents the system output where o is the number of outputs. The vector $x(t)$ of length n represents the system states which can be defined as a linear combination between past input and past outputs. This equation also contains state noise vector $w(t)$ representing the noise on the states (from outside perturbations) and the measurement noise vector $v(t)$.

The A matrix is the state matrix whose rank is the equivalent to the order of the system. The B matrix represents the effects of the inputs on each the state. The physical inputs of the system depend on the application and will be described in the second and third paper of this thesis. The C matrix relates the outputs to the states and represents the best linear combination of the states that form the output. The D matrix relates directly the inputs to the outputs without going through the state equation (2.2). A non-zero matrix D is equivalent to a system where some inputs influence the output without any time delay. In a state-space system, without the matrix D, the output $y(t)$ is a function of the state $x(t)$ which is a function of the input and state at the previous timestep (see equation 2.2).

2.2.2 Overview of the subspace system identification method

In order to define a valid linear model, it is necessary to obtain a proper estimate for the parameters in the matrices A, B, C and D. Many different approaches can be used to identify these parameters and it was decided to use a subspace method to perform this identification. The main advantage of the subspace algorithm used in this paper, is that it is a non iterative algorithm. While classical parameter estimation methods require a proper initial guess of the parameters in the $[A,B,C,D]$ matrices and minimize the error between the model and the data with a minimisation algorithm, the subspace identification method does not require such an initial guess and finds the value of the parameters using only the input and output data. The subspace identification method is therefore much faster than an iterative technique and does not suffer to problems related to optimisation such as possible convergence to local minimum. The subspace algorithm was implemented in Matlab® System identification toolbox. For basic understanding of the algorithm, the reader is invited to consult reference Ljung (1999) for the theory behind the algorithm and the toolbox user guide written by Ljung (2006) for how the algorithm was implemented in Matlab®. For further understanding, the reader can consult a very good application paper by Galvao (2005) where the subspace method is used in a fiber optic application and a good demonstration is done on some aspects of the algorithm. Plenty of authors have developed different aspects of the algorithm and they will be stated as these aspects are exposed.

The concept of the subspace method is to obtain the system observability matrix Γ_r defined in equation (2.4) from modern control theory. This matrix can be obtained using only the system inputs $u(t)$ and outputs $y(t)$.

$$\Gamma_r \overset{def}{=} \begin{bmatrix} C_{o\,x\,n} \\ CA_{o\,x\,n} \\ \cdots \\ CA^{r-1}{}_{o\,x\,n} \end{bmatrix}_{ro\,x\,n} \tag{2.4}$$

In equation (2.4) r represents a forward prediction horizon, the significance of this parameter will be explained in the next sections. Once this observability matrix Γ_r is known, it is possible to extract the state space matrices [A,B,C,D] using the input and output vectors. The detailed procedure to obtain the observability matrix and the discrete state space matrices now will be explained. The theory behind the subspace algorithm is divided into five sections. The first section defines basic matrices and equations necessary to the demonstration, sections 2 and 3 explain the two steps necessary to find the observability matrix of equation (2.4). Section 4 explains how to obtain the discrete state space matrices [A,B,C,D] from the observability matrix and, finally, section 5 explains how to obtain the state and noise vectors time histories (expressed in matrix form) of equations (2.2) and (2.3).

2.2.3 Basic definition of input and output matrices

In order to understand the subspace algorithm, it is necessary to define some important matrices. The input data which occur after a reference time t can be arranged into a *Hankel* matrix as follows:

$$U_f \overset{def}{=} \begin{bmatrix} u[t]_{m\,x\,1} & u[t+\Delta t]_{m\,x\,1} & u[t+2\Delta t]_{m\,x\,1} & \cdots & u[t+(j-1)\Delta t]_{m\,x\,1} \\ u[t+\Delta t]_{m\,x\,1} & u[t+2\Delta t]_{m\,x\,1} & u[t+3\Delta t]_{m\,x\,1} & \cdots & u[t+j\Delta t]_{m\,x\,1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ u[t+(r-1)\Delta t]_{m\,x\,1} & u[t+r\Delta t]_{m\,x\,1} & u[t+(r+1)\Delta t]_{m\,x\,1} & \cdots & u[t+(r+j-2)\Delta t]_{m\,x\,1} \end{bmatrix}_{rm\,x\,N} \tag{2.5}$$

In this matrix, the subscript f stands for *future* inputs because only the inputs occurring after the time t are included into the matrix. The subscript r, the forward prediction horizon used by the algorithm is the number of time steps used to build the *hankel* matrix U_f. This prediction horizon is chosen by the user and, according to Ljung (1999) the only requirement is that its value must be greater than the desired system order n. The subscript N represents the length of the measured output vector and the index j is a

dummy index with a value adjusted to insure that all data available in the identification are included on each line of the U_f matrix. Notice that if the expression in the brackets [] of the matrix element u has a value greater than N, the value of $u[]$ is zero. This is a property common to any *Hankel* matrix. Similar *Hankel* matrices can be constructed with the output vectors as follows:

$$Y_f \overset{def}{=} \begin{bmatrix} y[t]_{o\,x\,1} & y[t+\Delta t]_{o\,x\,1} & y[t+2\Delta t]_{o\,x\,1} & \cdots & y[t+(j-1)\Delta t]_{o\,x\,1} \\ y[t+\Delta t]_{o\,x\,1} & y[t+2\Delta t]_{o\,x\,1} & y[t+3\Delta t]_{o\,x\,1} & \cdots & y[t+j\Delta t]_{o\,x\,1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y[t+(r-1)\Delta t]_{o\,x\,1} & y[t+r\Delta t]_{o\,x\,1} & y[t+(r+1)\Delta t]_{o\,x\,1} & \cdots & y[t+(r+j-2)\Delta t]_{o\,x\,1} \end{bmatrix}_{ro\,x\,N} \tag{2.6}$$

The future state vector can be defined by use of the following matrix:

$$X_f = \begin{bmatrix} x[t]_{n\,x\,1} & x[t+\Delta t]_{n\,x\,1} & \cdots & x[t+(j-1)\Delta t]_{n\,x\,1} \end{bmatrix}_{n\,x\,N} \tag{2.7}$$

It is also necessary to define an extended controllability matrix as follows:

$$\Delta_r = \begin{bmatrix} \left(A^{r-1}B\right)_{n\,x\,1} & \left(A^{r-2}B\right)_{n\,x\,1} & \cdots & (AB)_{n\,x\,1} & B_{n\,x\,1} \end{bmatrix}_{n\,x\,r} \tag{2.8}$$

and, finally, the impulse response matrix given by:

$$H_r = \begin{bmatrix} D_{o\,x\,m} & 0 & 0 & 0 & 0 \\ CB_{o\,x\,m} & D_{o\,x\,m} & 0 & \cdots & 0 \\ CAB_{o\,x\,m} & CB_{o\,x\,m} & D_{o\,x\,m} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ CA^{r-2}B_{o\,x\,m} & CA^{r-3}B_{o\,x\,m} & CA^{r-4}B_{o\,x\,m} & \cdots & D_{o\,x\,m} \end{bmatrix}_{ro\,x\,Nm} \tag{2.9}$$

It is also important to describe the effect of noise on the *future* output of the system. This noise effect defined with the *Hankel* matrix V:

$$V \overset{def}{=} \begin{bmatrix} V_1[t]_{o\times 1} & V_1[t+1]_{o\times 1} & V_1[t+2]_{o\times 1} & \cdots & V_1[t+j-1]_{o\times 1} \\ V_2[t+1]_{o\times 1} & V_2[t+2]_{o\times 1} & V_2[t+3]_{o\times 1} & \cdots & V_2[t+j]_{o\times 1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ V_k[t+r-1]_{o\times 1} & V_k[t+r]_{o\times 1} & V_k[t+r+1]_{o\times 1} & \cdots & V_k[t+r+j-2]_{o\times 1} \end{bmatrix}_{ro\times N} \tag{2.10}$$

where the value of the term V_k is defined with equation (2.11):

$$V_k = CA^{k-2}w(t) + CA^{k-3}w(t+1) + \ldots + Cw(t+k-2) + v(t+k-1) \tag{2.11}$$

In the subspace algorithm, the *future* value of the output is related to the future value of the states and input by use of the following equations:

$$Y_f = \Gamma_r X_f + H_r U_f + V \tag{2.12a}$$

$$X_f = A^r X_p + \Delta_r U_p \tag{2.13b}$$

It is straightforward to show that equation (2.12) was derived from basic state-space equations (2.2) and (2.3). For example, with the maximum prediction horizon set to r = 2 and simplifying the system to a single input single output (SISO) system, the prediction equation (2.12a) can be simplified to the following equation:

$$\begin{aligned} Y_f &= \begin{bmatrix} y[t] & y[t+1] \\ y[t+1] & y[t+2] \end{bmatrix} = \begin{bmatrix} C \\ CA \end{bmatrix} \begin{bmatrix} x[t] & x[t+1] \end{bmatrix} + \\ &\begin{bmatrix} D & 0 \\ CB & D \end{bmatrix} \begin{bmatrix} u[t] & u[t+1] \\ u[t+1] & u[t+2] \end{bmatrix} + \begin{bmatrix} v[t] & v[t+1] \\ Cw[t]+v[t+1] & Cw[t+1]+v[t+2] \end{bmatrix} \end{aligned} \tag{2.13}$$

From equation (2.13), each term of the Y_f matrix can be expressed as:

$$Y_f(1,1) = y[t] = Cx[t] + Du[t] + v[t] \tag{2.14a}$$

$$Y_f(1,2) = y[t+1] = Cx[t+1] + Du[t+1] + v[t+1] \tag{2.14b}$$

$$Y_f(2,1) = y[t+1] = CAx[t] + CBu[t] + Du[t+1] + Cw[t] + v[t+1] \tag{2.14c}$$

$$Y_f(2,2) = y[t+2] = Cx[t+2] + Du[t+2] + v[t+2] \tag{2.14d}$$

It can be shown that equations (2.14a), (2.14b) and (2.14d) are the same as equation (2.3), equation (2.14c) is found by inserting equation (2.2) into equation (2.3). This step described the basic matrices used in the identification algorithm. These matrices will be used in the next two sections to obtain the observability matrix Γ_r. The next section will show how to remove the influence of future outputs matrix U_f and the noise matrix V on equation (2.12a) in order to isolate the observability matrix.

2.2.4 Determination of the states contribution to the output variables

Referring to equation (2.12a), the future value of the output Y_f is a function of the state and input vectors:

$$Y_f = \Gamma_r X_f + H_r U_f + V \tag{2.12a}$$

From the input and output data, it is possible to isolate the term $\Gamma_r X_f$ in equation (2.12a), this will be done to obtain the observability matrix Γ_r. The algorithm used to perform this separation is called *instrumental variables approach* and is thoroughly explained by Ljung (1999) and Viberg et al. (1995,1997). The following step will show how on orthogonal projection approach can be used to remove the U_f contribution to the output of equation 2.12a.

2.2.4.1 Orthogonal projection to remove U_f contribution

The first step for removing the input matrix contribution is to use a geometric interpretation of this equation as shown in figure 2.1 also explained by Galvao (2005).

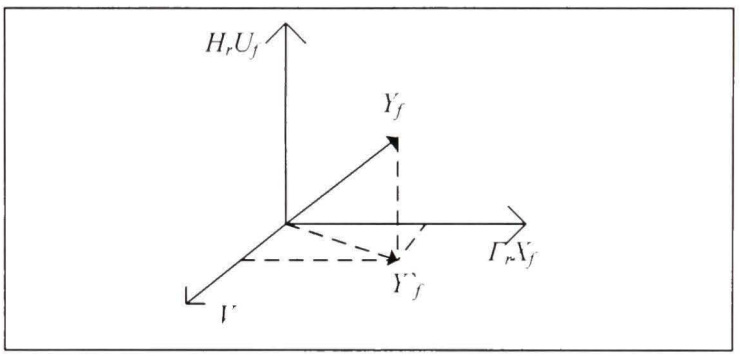

Figure 2.1 ***Perpendicular projection of the future outputs perpendicular to the future inputs***

If one interpret equation (2.12a) as a vector, the first step to isolate the observability matrix Γ_r is to project the output vector Y_f perpendicular to the input contribution $H_r U_f$. This can be done with the following projection matrix:

$$\left[\Pi_{U_f}^{\perp}\right]_{N \times N} = I - U_f^T \left(U_f U_f^T\right)^{-1} U_f \tag{2.15}$$

Where the superscript T means *transpose*. A detailed proof concerning this orthogonal projection matrix is available in the Galvao paper (2005). If this perpendicular projection operator is applied on U_f, it is equivalent to find the projection of $H_r U_f$ perpendicular to U_f which, logically, is zero. Mathematically, it can be shown by use of equation (2.16):

$$U_f \Pi_{U_f}^{\perp} = U_f - U_f U_f^T \left(U_f U_f^T\right)^{-1} U_f = U_f - U_f I = 0 \tag{2.16}$$

Post-multiplying both sides of equation (2.12a) by the projection operator $\Pi_{U^T}^{\perp}$ yields:

$$\left[Y_f \Pi_{U^T}^{\perp}\right]_{r \times N} = \Gamma_r X_f \Pi_{U^T}^{\perp} + H_r U_f \Pi_{U^T}^{\perp} + V \Pi_{U^T}^{\perp} = \left[\Gamma_r X_f \Pi_{U^T}^{\perp}\right]_{r \times N} + \left[V \Pi_{U^T}^{\perp}\right]_{r \times N} \quad (2.17)$$

The input contribution has now been removed from the prediction equation. The next step is to remove the noise contribution by using an instrument variable.

2.2.4.2 Removal of the noise contribution by use of an instrument variable

It is now necessary to exclude the noise term $V \Pi_{U^T}^{\perp}$. This can be done by post-multiplying equation (2.17) with suitable matrix Φ that is not correlated to the noise matrix. Let's define the Φ matrix as follows:

$$\Phi = [\varphi_s(t) \quad \varphi_s(t+1) \quad \ldots \quad \varphi_s(t+j-1)]_{s \times N} \quad (2.18)$$

where φ_s are vectors that are uncorrelated with the noise. The number of lines s of this matrix may have any value, but it has to be higher than the desired order n of the dynamic system. This variable is called an *instrument variable* as defined by Ljung (1999) and Viberg (1997) and it is used to reduce the noise term of equation (2.17) to zero. Let us post-multiply equation (2.17) by the transpose of Φ and normalize with the number of sample in the data as follows:

$$\left[G\right]_{ro \times s} \stackrel{def}{=} \frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T + \frac{1}{N} V \Pi_{U^T}^{\perp} \Phi^T \stackrel{def}{=} \left[\Gamma_r T_N\right]_{ro \times s} + \left[F_N\right]_{ro \times s} \quad (2.19)$$

In equation (2.19), the subscript N implies that the value of T_N and F_N are approximated for a data record containing N data points. In order to cancel out the noise term without

affecting the term dependant on the future states T_N, the requirement for a proper *instrument* matrix Φ is that it must be correlated to the future states X_f, but uncorrelated with the noise term, mathematically, it can be expressed by the following equations:

$$\lim_{N\to\infty} F_N = \lim_{N\to\infty} \frac{1}{N} V \Pi_{U^T}^{\perp} \Phi^T = 0 \tag{2.20}$$

$$\lim_{N\to\infty} T_N = \lim_{N\to\infty} \frac{1}{N} X_f \Pi_{U^T}^{\perp} \Phi^T = T \tag{2.21}$$

In equation (2.21), the parameter T is equivalent to the estimation of T_N with an infinite number of data points. Equation (2.20) implies that as the number of samples N goes to infinity, the noise matrix and the *instrument matrix* Φ must cancel each other. Intuitively, one can think that at each time step, the product of V and Φ^T will be either positive of negative and the sign of this product will vary in a random manner as these matrices are uncorrelated. For an infinitely large number of data points, the value of F_N will converge to zero. Equation (2.19) can therefore be summarized as:

$$G = \frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T = \Gamma_r T + E_N \tag{2.22}$$

where

$$E_N = \Gamma_r \left(T_N - T\right) + F_N \tag{2.23}$$

$$E_N \to 0 \text{ as } N \to \infty$$

In these equations, the term E_N is a measure of the estimation error of the model due the finite number of measured samples in the data set. This error goes to zero as the number of number of points in the data set goes to infinity.

The remaining step of the above demonstration is to find an appropriate *instrument* matrix Φ. As it was stated before, the first requirement for the *instrument* matrix Φ is that it must be correlated with the future states, which are unknown at this point. Even though these states are unknown, their value is dependant on the past inputs and outputs, these past inputs and outputs are therefore a possible choice. The second requirement is that the matrix Φ must be uncorrelated with the noise. This is always the case when the system inputs are properly selected. This is because the noise can be seen as the error between the model and the data and, for a good model, this error is completely random. The past input and outputs are therefore a logical choice for the *instrument* matrix. In this paper, the *instrument* matrix used is described in equation (2.24) which was taken from Ljung (1999).

$$\Phi = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}_{s \times N} = \left[\begin{array}{ccccc} y[t-1]_{o\times 1} & y[t]_{o\times 1} & y[t+1]_{o\times 1} & \cdots & y[t+j-2]_{o\times 1} \\ y[t-2]_{o\times 1} & y[t-1]_{o\times 1} & y[t]_{o\times 1} & \cdots & y[t+j-3]_{o\times 1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ y[t-h]_{o\times 1} & y[t-h+1]_{o\times 1} & y[t-h+2]_{o\times 1} & \cdots & y[t+h+j-1]_{o\times 1} \\ \hline u[t-1]_{m\times 1} & u[t]_{m\times 1} & u[t+1]_{m\times 1} & \cdots & u[t+j-2]_{m\times 1} \\ u[t-2]_{m\times 1} & u[t-1]_{m\times 1} & u[t]_{m\times 1} & \cdots & u[t+j-3]_{m\times 1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ u[t-h]_{m\times 1} & u[t-h+1]_{m\times 1} & u[t-h+2]_{m\times 1} & \cdots & u[t+h+j-1]_{m\times 1} \end{array}\right]_{s\times N} \qquad (2.24)$$

In this equation, the subscript p stands for *past* input and outputs and the parameter h is the number of past inputs and outputs used by the algorithm. The number of past inputs h is a parameter chosen by the user. A higher value of h will lead to a better fit, but choosing very high number of past inputs cannot be done in practice because limitations in the length of the data set. Ljung (1999) proposed to limit the value of h to the optimal prediction horizon that would be used if the model was identified with an ARX structure. This optimum choice is made based on Akaike Information Criterion defined by Akaike (1974). This criterion is widely used to compromise between the quality of a

fit and the complexity of a model. The parameter h used in this paper was chosen based on the above method.

Once a proper instrument matrix has been found, assuming there is enough data points in the record to bring the error due to noise E_N to 0, equation (2.22) reduces to:

$$[G]_{\text{ro x s}} = [\Gamma_r]_{\text{ro x n}} [T]_{\text{n x s}} \tag{2.25}$$

And, using equation (2.19), equation (2.25) becomes:

$$\frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T \tag{2.26}$$

At this point, every term of the left hand side of equation (2.26) are known. The next step will be to extract the observability matrix Γ_r from equation (2.26). This can be done by performing a Singular Value Decomposition (*SDV*). More details on this procedure will be explained in section 3.

2.2.5 Determination of the observability matrix from singular value decomposition

Once the matrix G is known from equation (2.25), it is possible to decompose it into three sub-matrices using a well known linear algebra theorem called *Singular Value Decomposition*. This theorem is explained in through details in reference by Patel et al. (1993). This decomposition is expressed as follows:

$$G_{\text{ro x s}} = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T = U_{\text{ro x ro}} S_{\text{ro x s}} V_{\text{s x s}}^T \tag{2.27}$$

In equation (2.27), the matrix S is made of the singular values of G. These singular values are the positive square root of the eigenvalues of $\left(G^T G\right)$, these eigenvalues are sorted in descending order form the first row to the last row of matrix S. As it will be illustrated later on, the goal of doing a singular value decomposition is to separate the contribution of the observability matrix from the contribution of the states to the term G. Equation (2.27) may also be read as:

$$G_{\text{ro x s}} = U_{\text{ro x ro}} \begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_s \\ 0 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}_{\text{ro x s}} V^T_{\text{s x s}} \tag{2.28}$$

In equation (2.28), the parameters σ_i represent the singular values of the matrix G. The matrices U and V are called singular vectors. They are respectively the orthonormal eigenvectors of $\left(GG^T\right)$ and of $\left(G^T G\right)$. Since these eigenvectors are orthonormal, they agree with the following equations:

$$\begin{aligned} U^T U &= I_{\text{ro x ro}} \\ V^T V &= I_{\text{s x s}} \end{aligned} \tag{2.29}$$

where the matrix I is the identity matrix. The following demonstration will show how the new matrices defined by the *Singular Value Decomposition* can be used to obtain an extended observability matrix Γ_r that relates properly the inputs to the outputs. The singular value decomposition of equation (2.27) applies for a dynamic system of any order. If the desired order n of the dynamic system is known, a proper procedure

separate the first n singular values and singular vectors of the system from the other singular values and vectors. Equation (2.27) therefore becomes:

$$G_{\text{ro x s}} = \begin{bmatrix} U_1 & | & U_2 \end{bmatrix}_{\text{ro x ro}} \begin{bmatrix} S_1 & | & S_2 \\ - & - & - \\ S_3 & | & S_4 \end{bmatrix}_{\text{ro x s}} \begin{bmatrix} V_1^T \\ - \\ V_2^T \end{bmatrix}_{\text{s x s}} \tag{2.30}$$

Here, equation (2.30) is equivalent to equation (2.27), but is separated into different components. The U_1 matrix is now an ro x n matrix, the matrix S_1 is a n x n matrix and the matrix V_1 is a s x n matrix. The approximation of G matrix for a system with a certain order n therefore becomes:

$$G\_approx_{\text{ro x s}} = U_1 S_1 V_1^T = \Gamma_r T = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T \tag{2.31}$$

where G_new is the new approximated matrix G. Notice that G has the same size as before, however, the size of U_1 is ro x n, the size of S_1 is n x n and the size of V_1 is s x n. When this approximation is performed, a certain number of singular values and singular vectors are dropped. Since the singular values and vectors of equation (2.27) were placed in a descending order, only the ones with a smaller value are dropped. The singular values that were kept represent the true dynamic of the system and the small singular values that are not taken into account represents errors due to noise. If the order of the system n is chosen to be too low, the model will not match the data properly because some significant dynamic will be lost. If the selected order is too high, this may lead to the problem of *overfitting* as described by Ljung (1999) and many other authors. Overfitting is what happens when the order of the equations representing a dynamic system are higher than the true order of the system. In this case, the system identification algorithm attempts to match the noise contribution with the remaining free parameters of

the linear equations (2.2) and (2.3). This usually results in a model that gives good results on the data used to create it, but poor results on validation data.

Once the new matrix G_approx has been determined, it is possible to obtain a proper estimate of the observability matrix Γ_r. Notice that many different combinations of observability matrix and T matrix can lead to a set of parameters value that insures a proper match. By inspection of equation (2.31), we notice that the size of the matrix U_1 is the same as the size of matrix Γ_r, we also observe that the size of the matrix V_1^T is the same as the size of the matrix $\tilde{T}$. This can be summarised in the following equation (2.32):

$$\begin{aligned} G\_approx_{\text{ro x s}} &= U_{1(\text{ro x n})} S_{1(\text{n x n})} \left(V_1^T\right)_{(\text{n x s})} \\ G\_approx_{\text{ro x s}} &= \Gamma_{r(\text{ro x n})} T_{(\text{n x s})} \end{aligned} \tag{2.32}$$

According to reference Ljung (1999), it follows that possible values of the observability matrix $\Gamma_{r(\text{r x n})}$ may be:

$$\Gamma_r = U_1 \tag{2.33}$$

It is also possible to add weight functions to the matrix G obtained from equation (2.27) before performing singular value decomposition as follows:

$$G_{Wt} = W_1 G W_2 \tag{2.34}$$

The reason for adding weight matrices is to remove any residual error due to noise (E_N term, see equation (2.23)) in the variable G. These weight matrices are made of parameters that are uncorrelated with the projected noise matrix F_N. In the absence of

noise, adding a weight matrix has no effect on the identification results, but it improves the results when the identification is done on noisy data.

After a weight matrix is selected, the new observability matrix is found by use of the following equation:

$$\Gamma_r = W_1^{-1} U_1 \tag{2.35}$$

Where the matrices U_1 has been found by performing a singular value decomposition on G_{W1} found in equation (2.35). Many authors have proposed expressions for different weight matrices and they have an influence on the results of the identification. A good summary on the work of different authors on these weight matrices formulations has been done by Viberg (1997) where different expressions for weight matrices are derived using the same mathematical approach. For original work on the different weight matrices, the reader is invited to consult the papers written by Verhaegen (1994), Van Overschee (1994), Viberg (1995) and Larimore (1990). In this thesis, the weight matrices defined by Larimore (1990) give excellent results. Since there is not much noise in the data related to this project, the algorithm is not very sensitive to the selected weight and it wasn't necessary to try other weight formulations to obtain good results. The weights as defined by Larimore are defined by use of the following equation:

$$W_1 = \left(\frac{1}{N} Y \Pi_{U^T}^{\perp} Y^T \right)^{-1/2} ; W_2 = \left(\frac{1}{N} \Phi \Pi_{U^T}^{\perp} \Phi^T \right)^{-1/2} \tag{2.36}$$

At this point the observability matrix Γ_r have been determined from equation (2.35). This observability matrix can now be used to obtain the value of the matrices [*A, B, C, D*]. The procedure to do it will be shown in the next section.

2.2.6 Determination of the matrices describing the model by using the observability matrix Γ_r and estimation of a state sequence

This section will first explain how to find the system matrices A and C defined in equation (2.2) and (2.3) by using the observability matrix Γ_r. The matrices related to the system's inputs B and D will then be found by use of a linear regression. Finally, the system matrices will be used to obtain the system's state vector $\hat{x}$. This state vector will be used later to find the noise vector v and w.

2.2.6.1 Estimating A and C matrices from the observability matrix Γ_r

Once the observability matrix is known, it is quite easy to obtain the estimates of the A and C matrices. If we refer to equation (2.4) as it was defined at the beginning of the theory section:

$$\Gamma_r \overset{def}{=} \begin{bmatrix} C_{o \times n} \\ CA_{o \times n} \\ \cdots \\ CA^{r-1}{}_{o \times n} \end{bmatrix}_{ro \times n} \tag{2.4}$$

The estimate matrix C is obtained by taking the following terms of the observability matrix:

$$\hat{C} = \Gamma_{(1:o,1:n)} \tag{2.37}$$

Where the hat "^" means that it is an estimate. Equation (2.37) simply means that we have to extract the first o lines and n columns of the observability matrix. The matrix A can be found from the observability matrix by solving the following equation:

$$\Gamma_{(o+1:ro,1:n)} = \Gamma_{(1:o(r-1),1:n)} A \tag{2.38}$$

In this equation, the left hand side represents the observability matrix with the first sub-matrix $\hat{C}$ removed and the right hand side represents the observability matrix with the last sub-matrix CA^{r-1} removed. This is equivalent to the following equation:

$$\begin{bmatrix} CA_{o\,x\,n} \\ CA^2_{o\,x\,n} \\ \cdots \\ CA^{r-1}_{o\,x\,n} \end{bmatrix}_{(r-1)o\,x\,n} = \begin{bmatrix} C_{o\,x\,n} \\ CA_{o\,x\,n} \\ \cdots \\ CA^{r-2}_{o\,x\,n} \end{bmatrix}_{(r-1)o\,x\,n} A_{n\,x\,n} \tag{2.39}$$

In this equation, the only unknown is the state matrix A. If the modified observability matrices $\Gamma_{(o+1\,ro,1:n)}$ and $\Gamma_{(1:o(r-1),1:n)}$ of equation (2.38) were both square matrices, it would be easy to find $\hat{A}_{n\,x\,n}$ by just pre-multiplying both side of the equation by the inverse of $\Gamma_{(1:o(r-1),1:n)}$. Since these matrices are not necessarily square, equation (2.38) can be solved by use of the following equation:

$$\hat{A}_{n\,x\,n} = \left(\begin{bmatrix} C_{o\,x\,n} \\ CA_{o\,x\,n} \\ \cdots \\ CA^{r-2}_{o\,x\,n} \end{bmatrix}^{\dagger} \right)_{n\,x\,(r-1)o} \begin{bmatrix} CA_{o\,x\,n} \\ CA^2_{o\,x\,n} \\ \cdots \\ CA^{r-1}_{o\,x\,n} \end{bmatrix}_{(r-1)o\,x\,n} \tag{2.40}$$

In equation (2.40), the superscript "$\dagger$" denotes the Moore-Penrose pseudo-inverse described by Viberg (1995). This is a more general type of inversion which does not require the matrices to be square. The pseudo-inverse can be computed by singular value decomposition and reader is invited to consult references from Patel (1993) and Klema (1980) for more details on this operation.

2.2.6.2 Estimating B and D matrices from linear regression

Once the system matrices $\hat{A}$ and $\hat{C}$ are known, it is possible to estimate the $\hat{B}$ and $\hat{D}$ matrices by use of a linear regression technique. At this point, the output y can be related to the input u by converting equations (2.2) and (2.3) into a discrete transfer function. Using the following discrete operator z:

$$\begin{aligned} zx(t) &= x(t+\Delta t) \\ z^{-1}x(t) &= x(t-\Delta t) \end{aligned} \tag{2.41}$$

Equation (2.2) becomes:

$$zx(t) = \hat{A}x(t) + Bu(t) + w(t) \Rightarrow x(t) = \left(zI - \hat{A}\right)^{-1} Bu(t) + \left(zI - A\right)^{-1} w(t) \tag{2.42}$$

Inserting equation (2.42) into equation (2.3) and neglecting the noise contribution $w(t)$ and $v(t)$ yields:

$$\hat{y}(t \mid B, D) = \hat{C}\left(zI - \hat{A}\right)^{-1} Bu(t) + Du(t) \tag{2.43}$$

where $\hat{y}(t \mid B, D)$ means the *estimated* y conditional to B and D. A very efficient way to find the unknown parameters B and D of equation (2.44) is to use a linear regression method. From linear regression theory, as explained by Ljung (1999) the estimated output $\hat{y}(t)$ may be expressed by the following equation:

$$\hat{y}(t)_{o \times 1} = \eta(t)\hat{\theta} = \eta(t)_{o \times (nm+om)} \begin{bmatrix} Vec(B) \\ Vec(D) \end{bmatrix}_{(nm+om) \times 1} \tag{2.45}$$

In equation (2.45), the matrix $\eta(t)$ is made of the past and presents inputs $u(t)$ of the system the procedure to construct this matrix will be explained later on. The single column vector θ represents all the unknown parameters to be found, these unknown parameters are all the elements of the matrices B and D. The operator "Vec" builds a column vector from a matrix by stacking its columns on top of each other. Recall that the index o represents the number of outputs, n is the order of the system and the number of states and m is the number of inputs to the system.

The estimated parameters $\hat{\theta}$ equation (2.45) can be found be formulating it as a least square problem as follows:

$$J\left(\hat{\theta}\right)=\frac{1}{N}\sum_{t=1}^{N}\left(y(t)-\eta^{T}(t)\hat{\theta}\right)^{2} \tag{2.46}$$

Where $J\left(\hat{\theta}\right)$ is the error between the model output and the output from the flight test data and N is the total number of samples of inputs and outputs. The minimum error can be found without iteration by use of the following equation from linear regression theory as described by Ljung (1999):

$$\hat{\theta}_{N}=\left[\sum_{t=1}^{N}\eta(t)\eta^{T}(t)\right]^{-1}\sum_{t=1}^{N}\eta(t)\hat{y}(t) \tag{2.47}$$

In equation (2.47), $\hat{\theta}_{N}$ represents the best estimate of the parameter vector using N samples of inputs and outputs. What remains to be done is to define the matrix $\eta(t)$, the best way to illustrate how to construct the matrix $\eta(t)$ is with an example. Suppose that we have a second order 2 inputs and 1 output state space system, equation (2.43) reduces to:

$$\hat{y}(t \mid B, D) = \hat{C}\left(zI - \hat{A}\right)^{-1} Bu(t) + Du(t) = \hat{C}\left(zI - \hat{A}\right)^{-1} \begin{bmatrix} \theta_1 & \theta_3 \\ \theta_2 & \theta_4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} \theta_5 & \theta_6 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{2.48}$$

Before solving the least square problem, it is necessary to convert equation (2.48) into the format used in equation (2.45), this is illustrated in equation (2.49):

$$\hat{y}(t)_{0 \times 1} = \eta(t)\hat{\theta} = \begin{bmatrix} \beta \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_1 & \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_1 & \beta \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_2 & \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_2 & u_1 & u_2 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \dots \\ \theta_6 \end{bmatrix} \tag{2.49}$$

where

$$\beta = \hat{C}\left(zI - \hat{A}\right)^{-1} \tag{2.50}$$

Once the parameters $\hat{\theta}_N$ have been determined, it is possible to reconstruct the matrices B and D using these parameters. At this point, the system matrices have all been determined. Once the system matrices A, B, C, D are known, it is also possible to determine the measurement noise $v(t)$ and state noise $w(t)$ vector from input and output data and a noise covariance matrix describing the relationship between the output noise and the state noise. This will be explained in the following section.

2.2.7 Estimation of the state and noise matrices

Once the state space matrices [A, B, C, D] have been determined, it can be useful to obtain information about the state noise w and output noise v matrices which are defined in equations (2.2) and (2.3). In order to obtain these matrices, is necessary to estimate the state vector $\hat{x}$. The following section will describe the procedure to estimate the state vector, the output noise vector and the state noise vector.

2.2.7.1 Estimating the state vector by singular value decomposition

In equation (2.51), as in equation (2.26) after the input and noise contributions were removed from the output, the system output and states vector were related by the following equation:

$$G = \frac{1}{N} Y_f \Pi^{\perp}_{U^T} \Phi^T = \frac{1}{N} \Gamma_r X_f \Pi^{\perp}_{U^T} \Phi^T \tag{2.51}$$

And recall that the matrices that are used in this equation have the following size:

$$G_{[\text{ro x s}]};\ Y_{f\,[\text{ro x N}]};\Pi^{\perp}_{U^T\,[\text{N x N}]};$$
$$\Phi_{[\text{s x N}]};\Gamma_{r[\text{ro x N}]};\ X_{f\,[\text{n x N}]};$$

The states matrix X_f represents the system states at any given time where n is the number of states and N is the number of time samples of the data record. In order to find this matrix, is necessary to remove the term $\Pi^{\perp}_{U^T} \Phi^T$ of equation (2.51), which is done by post-multiplying it by use of following weights on G matrix:

$$G_{Wt} = W_1 G W_2 \tag{2.52}$$

where

$$W_1 = I$$
$$W_2 = \left(\Phi \Pi^{\perp}_{U^T} \Phi^T\right)^{-1} \Phi$$

Inserting equation (2.52) into equation (2.51) yields:

$$G_{Wt} = \frac{1}{N} I Y_f \Pi^{\perp}_{U^T} \Phi^T \left(\Phi \Pi^{\perp}_{U^T} \Phi^T\right)^{-1} \Phi = \frac{1}{N} I \Gamma_r X_f \Pi^{\perp}_{U^T} \Phi^T \left(\Phi \Pi^{\perp}_{U^T} \Phi^T\right)^{-1} \Phi \tag{2.53}$$

This equation can be simplified as follows:

$$G_{Wi} = Y_f \Pi_{U^T}^{\perp} \Phi^T \left(\Phi \Pi_{U^T}^{\perp} \Phi^T \right)^{-1} \Phi = \Gamma_r X_f \tag{2.54}$$

From this equation, the state matrix can be found by performing a *singular value decomposition* of the matrix G_{Wi} as shown in the following equation:

$$G_{Wi} = \Gamma_r X_f = U_1 S_1 V_1^T \tag{2.55}$$

where U_1, S_1 and V_1 were defined in equation (2.32). From equation (2.54), the state sequence matrix X_f may be found with the following equation:

$$X_f = S_1 V_1^T \tag{2.56}$$

This also confirms equation (2.33) in section 2.2.5 which is shown below:

$$\Gamma_r = U_1 \tag{2.33}$$

The state sequence matrix has now been properly determined.

2.2.7.2 Estimation of the noise vector in time

At this point, every term of equation (2.2) and (2.3) have been determined. It can also be interesting to estimate the system noise vector in time, which is done by simply rearranging equations (2.2) and (2.3) into the following equations:

$$w(t) = x(t + \Delta t) - Ax(t) - Bu(t) \tag{2.57}$$

$$v(t) = y(t) - Cx(t) - Du(t) \tag{2.58}$$

where the only unknowns are the noise terms on the left hand side. The state noise $w(t)$ and the output noise $v(t)$ are related by the following equation:

$$w(t) = Kv(t) \tag{2.59}$$

Where K is called the noise covariance matrix and can be estimated by a least square procedure.

CHAPTER 3

INTRODUCTION TO THE FIRST PAPER

The following paper, as explained in the previous chapters, shows the formulation of a ground dynamics model for a B-427 helicopter. The structure of this model can be said to be a grey *box* model and most of challenges related to this paper were to determine a proper model structure.

This model was successfully implemented in a level D complete flight model and passed the *Proof of Match* validation process as defined by the FAA advisory circular AC-120-63 for a helicopter simulator qualification.

Under print in the following journal (page proofs accepted):

1-Nadeau Beaulieu M., Botez, R. M., Hiliuta A., 2006, *Ground dynamics Model Validation by Use of Landing Flight Test*, AIAA Journal of Aircraft, July 2006

Presented to the following conferences:

1-Nadeau-Beaulieu M., Botez R.M., 2006, *Ground dynamics model validation by use of landing flight test data*, AIAA Modeling and simulation technologies conference exhibit, Keystone, Colorado, August 21-24 2006.

2-Nadeau-Beaulieu M., Botez R.M., Hiliuta A., Popov A-V., Mureithi N., 2006, *Validation of a Ground dynamics model formulation by use of landing data*, International Symposium on Industrial Electronics ISIE, École de technologie supérieure, Montréal, July 9-13 2006. **(Best Oral Presentation Award)**

3- Nadeau Beaulieu M., Popov A-V., Hiliuta A. Botez R., Seto, J., Lambert, E., Mureithi, N, 2005, *New Formulation of the Ground Dynamics Model Generation and Validation by use of Flight Test Data*, Eleventh Australian International Aerospace Congress, Melbourne, Australia, March 13-17 2005

CHAPTER 4

GROUND DYNAMICS MODEL VALIDATION BY USE OF LANDING FLIGHT TEST

Nadeau Beaulieu M.[1], Botez R.M.[1], Hiliuta A.[1]

[1] Laboratory of Active Control, Avionics and Aeroservoelasticity, École de technologie supérieure, 1100, rue Notre-Dame Ouest, Montréal, Québec, Canada, H3C 1K3

4.1 Abstract

In this paper, a new formulation for a ground dynamics model of a commercial two-engine helicopter is validated after touchdown. The inputs of the ground dynamics model are the velocities and angles at touchdown, and its outputs are the forces and moments produced by the ground on the helicopter. Expressions for forces and moments, which depend on the ground contact force, the friction coefficient between the skids and the ground, and the system stiffness and damping are determined. The system stiffness and damping are defined between the touchdown point and the center of gravity in each of the helicopter's degree of freedom. Expressions for the stiffness, damping and friction coefficients are validated for two kinds of landing situations: one-engine inoperative and autorotation. The ground dynamics of the Bell-427 helicopter model are then used to build and certify a level-D flight simulator.

4.2 Introduction

4.2.1 Objectives

The research presented in this paper is part of a broader project, *Development of Global Model Parameter Estimation Technology*. In this project, a global flight dynamics helicopter model is developed and implemented in a Level D flight simulator and is validated by use of flight test data.

This paper presents the ground dynamics part of the helicopter flight simulator model. The ground dynamics model describes the helicopter motion after touchdown. At touchdown, the impact of the helicopter's skids with the ground introduces forces and moments. The moments are calculated between the helicopter's center of mass and the touchdown point on the skids and arise from the oscillations of the helicopter structure and the friction forces between the helicopter skids and the ground. The aim of this mathematical model is the calculation of the linear and angular accelerations felt by the pilot in the cockpit, rather than all of the details of the physical interactions between the ground, the helicopter skids and its fuselage. The helicopter model validation was realized by comparing the helicopter's orientation angles and ground speeds time histories with the landing data time histories. The FAA (Federal Aviation Administration) tolerances for One Engine Inoperative (OEI) landing and autorotation landing cases were respected in both cases. An autorotation landing occurs when the helicopter's engines are inoperative. In this case, the main rotor rpm and thrust are maintained by the airflow oriented upwards when the helicopter descends. Our general mathematical model will improve understanding of the helicopter's behaviour during touchdown and the principles it uses could be used to study the landing of another helicopter.

4.2.2 Literature Review

Two types of bibliographical research reviews are presented in Sections 4.2.2 and 4.2.3.

4.2.2.1 Collision and contact rigid body theories

There are only a limited number of publications in the field of helicopter ground dynamics. For this reason, it is necessary to refer to rigid body collision and contact theories in order to find theoretical formulations of an impact between an object and the ground. Three existing methods can be used in touchdown modeling and in the contact

forces calculations between the helicopter and the ground, and they are: the *Impulse-constraint method*, the *Impulse method* and the *Penalty method*.

The *impulse-constraint method* was developed first by Baraff (1989), and it uses an instantaneous impulse to model the impact between a rigid body and the ground. The magnitude of this impulse depends on the restitution coefficient values. The relative normal velocity of an object $u_{rel,N}$ colliding with the ground can be expressed with equation (4.1):

$$u^{+}_{\mathrm{Re}l,N} = -\varepsilon u^{-}_{\mathrm{Re}l,N} \tag{4.1}$$

where $u^{+}_{\mathrm{Re}l,N}$ is the relative normal velocity after touchdown, $u^{-}_{\mathrm{Re}l,N}$ is the relative normal velocity before touchdown and ε is the restitution coefficient.

The magnitude of the impulse applied on the helicopter depends on its mass, inertia and restitution coefficient values. Following touchdown, when the helicopter stays on the ground, the analytical constraint forces are tangential to the ground and these forces are solved analytically. This method is computationally expensive as a high number of iterations are necessary to obtain the constraint forces.

The *impulse method* is a variant of the impulse-constraint method. The touchdown of an object with the ground is modeled by an impulse; however, when the object is in contact with the ground, no constraint forces are applied and the object motion is stopped by use of multiple impulses. The magnitudes of the impulses are chosen so that the ground restitution coefficient is zero. This method is easier to implement as it does not require iterations, but it is less physically exact than the impulse-constraint method. However, the impulse method is computationally expensive when the object has more than one contact point with the ground and, therefore, is subjected to a high number of impulses.

The method was further improved by Guendelman (2003) with the use of lateral impulses to calculate the static and dynamic friction coefficients.

The third method, called the *penalty method*, is, by far, the simplest to implement. This method consists of modeling the ground as a spring and letting the object penetrate into the ground. At the moment when the object penetrates into the ground, a vertical force pushes this object back upwards. This force is proportional to the distance with which the object traveled into the ground. For collision studies, this method is less physically exact than the impulse method. However, the impulse method must evaluate the exact collision time between an object and the ground in order to be accurate, while the penalty method does not require this type of evaluation. The first two methods have never been applied to helicopters.

4.2.2.2 Application to an unmanned helicopter

The penalty method was used by Johnson (1997) to model the ground dynamics of an unmanned helicopter. Johnson used the penalty method because the small ground penetration present in the penalty method can be interpreted in this case as skid deformation. The helicopter was modeled by use of three-dimensional springs at each helicopter touchdown point.

In Johnson's model, when the helicopter was on the ground, a dynamic friction force was considered in cases when the total force applied on any on the lateral springs exceeds the maximum static friction force or the lateral velocity of any of these springs was non-zero.

Johnson did not specify how the stiffness and damping of each spring was selected. This unmanned helicopter model was not validated by use of landing test data.

4.3 Methodology

4.3.1 Coordinates system definitions

In the Earth-axis coordinate system, the x-axis points toward the North, the y-axis points toward the East and the z-axis points downwards (towards the Earth's center). The body-axis coordinate system is attached to the helicopter and is defined in figure 4.1

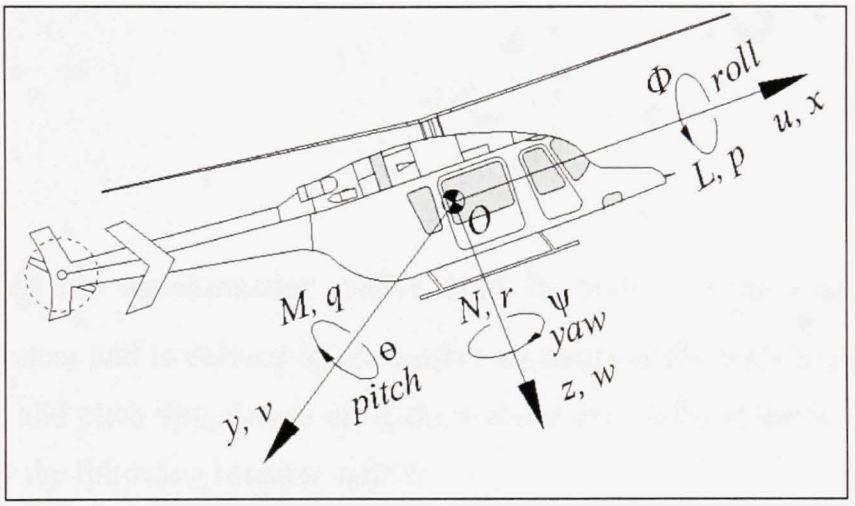

Figure 4.1 ***Body-axis system of a helicopter where x,y,z are the axes, u,v,w are the linear velocities, p,q,r are the angular velocities, ϕ,θ,ψ are the Euler angles, F_x,F_y,F_z are the forces and L,M,N are the rolling, pitching and yawing moments.***

In the ground dynamics model, a modified Earth axis coordinates system was used. In this coordinate system, the z-axis points down with respect to the earth and the x and y axes are parallel to the ground however, the axes are allowed to rotate with the helicopter in yaw. This implies that the helicopter x-axis points to the longitudinal direction of the helicopter motion with respect to the ground and the y-axis points toward the lateral direction of the helicopter motion with respect to the ground. This coordinate system definition is the most convenient one since the normal force between the ground and the helicopter is in the vertical direction and the friction forces are exerted in the x and y directions in the path of the helicopter motion with respect to the ground. This coordinate system also insures that the rolling and pitching moments on the

helicopter after the touchdown are not coupled with the yaw angle. These forces and moments are then converted into the body-axis coordinate system because this set of coordinates is used in the global simulation. The position $[x \quad y \quad z]_b^T$ of a point on the helicopter in the body-axis coordinate system can be converted to the modified Earth-axis coordinate system $[x \quad y \quad z]_e^T$ by use of the following transformation.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_{em} = [T]_{emb} \begin{bmatrix} x \\ y \\ z \end{bmatrix}_b \tag{4.2}$$

where $[T]_{emb}$ is the transformation matrix from the body's to the modified Earth's coordinates system and is defined by successive rotations of the body axes coordinates system in roll and pitch direction to bring the x and y axes parallel to the earth. This is represented by the following rotation matrix:

$$[T]_{emb} = \begin{bmatrix} \cos\theta & \sin\phi\sin\theta & \cos\phi\sin\theta \\ 0 & \cos\phi & -\sin\phi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \tag{4.3}$$

This transformation matrix $[T]_{emb}$ is a function of the helicopter's roll and pitch angles angles $[\phi \quad \theta]$.

The rates of change in time of the Euler angles $[\dot{\phi}, \dot{\theta}, \dot{\psi}]$ are further calculated as a function of the angular rates (p,q,r) in the body axis as defined in many textbooks such as Nelson (1998):

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & -\cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & -\sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{4.4}$$

4.3.2 Global Model Simulation Structure

The implementation of the ground dynamics model in the global simulation model is diagrammed in figure 4.2.

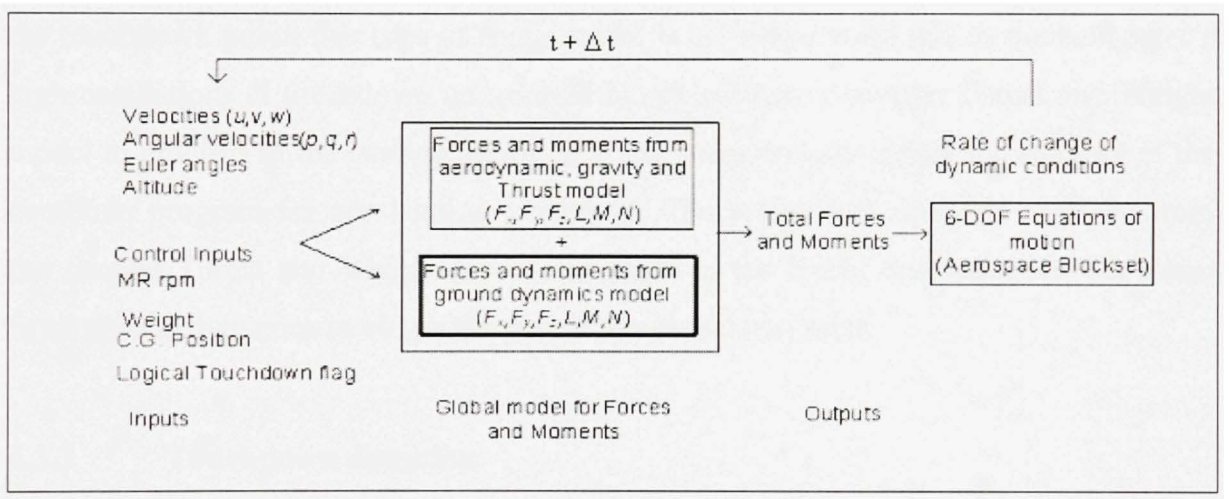

Figure 4.2 ***Implementation of the ground dynamics model in the global simulation***

As shown in figure 4.2, the global model for force and moment calculations in time uses inputs related to the helicopter flight conditions, and produces the forces and moments applied on the helicopter. These forces and moments are then inputs to the 6 degree-of-freedom helicopter equations of motion and the rates of change in its dynamic conditions (velocities, angular velocities, Euler angles and altitudes) are obtained. These rates are used to update the linear and angular velocities, the Euler angles and the altitudes for the next time step ($t + \Delta t$). The last input in figure 4.2 is a logical touchdown flag that is used to activate the ground dynamics model. This logical flag depends on the altitude of the lowest point on the helicopter skids (see Section 4.3.3). The global forces and

moments model includes two calculation parts: (1) for the helicopter's aerodynamic, thrust and gravity forces and moments, and (2) for the helicopter's forces and moments on the helicopter from ground reaction, friction, and oscillation stiffness and damping (known as the ground dynamics model).

When the helicopter flies above the ground, the forces and moments from the ground dynamics model are zero and the forces and moments from aerodynamics, gravity and thrust are computed by use of a flight model based on stability and control derivatives that are based on flight test data and validated by parameter estimation techniques. At the touchdown point, this type of flight model is no longer valid due to the helicopter's high oscillations at touchdown and should be replaced by a simpler Thrust and Weight model in order to fit the landing data and, at the same time, to ensure the stability of the computer program for any landing conditions. The forces and moments obtained from the simpler Thrust and Weight model are added to the forces and moments calculated from ground dynamics to obtain the global forces and moments.

4.3.3 Touchdown detection

During the helicopter's simulation, it is necessary to detect when the touchdown occurs in order to activate the ground dynamics model. The touchdown point is determined by computing the vertical distance between the ground and the lowest point on the helicopter skids in the modified Earth-axis coordinate system. The altitude of the lowest point on the helicopter is computed with equation (4.5):

$$Alt_{lowest\_pt} = Alt_{C.G.} - z_{e,lowest} \tag{4.5}$$

where Alt_{lowest_pt} is the unknown altitude of the lowest point on the skids. The altitude $Alt_{C.G.}$ is the altitude of the helicopter's center of gravity. This altitude is calculated during the simulation, from the $Alt_{C.G.}$ at the previous time step. In the landing data, this

altitude is given by the altimeter reading, corrected to ensure that the altitude equals the vertical CG position $z_{C.G}$ when the helicopter is on the ground. The distance $z_{e,lowest}$ is calculated from the skid geometry and the helicopter Euler angles. The relationship between the altitude of the helicopter's lowest point and the center of gravity altitude is illustrated in figure 4.3:

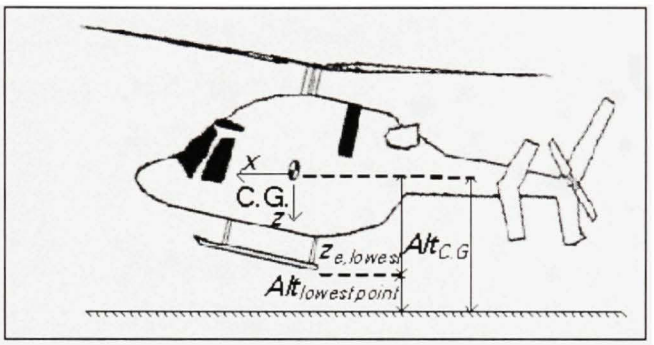

Figure 4.3 ***Definition of the altitude of the lowest altitude point on the helicopter***

The coordinates of the skid's extremities with respect to the center of gravity of the helicopter in the body-axis coordinate system may be found. These coordinates are converted from the body-axis into the modified Earth-axis coordinate system and the vertical distance between the lowest skid point and the center of gravity is denoted as $z_{e,lowest}$. Once $z_{e,lowest}$ is calculated, we can then calculate Alt_{lowest_pt}. Details of the skid geometry and calculation of the coordinates of the skids' extremity in the body-axis are found in the BHT Customer Operation Manual (see figure 4.4 and equation 4.6).

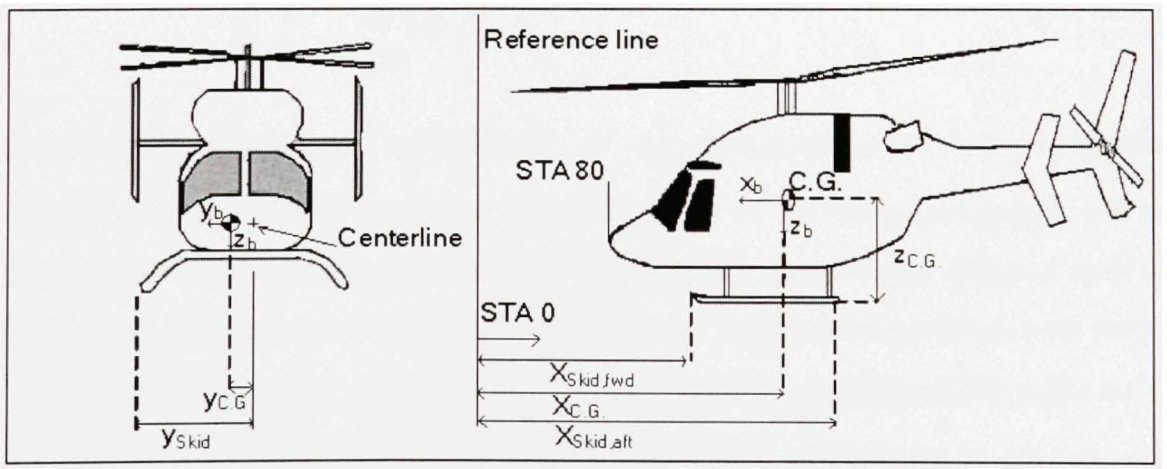

Figure 4.4 ***Measurements of the distance between the skids extremities and helicopter's CG (2 views)***

From figure 4.4, the coordinates $[x_b, y_b, z_b]$ of each skid extremity with respect to the center of gravity in the body-axis coordinate system can be found using equations (4.6.1) to (4.6.4).

Aft right skid extremity

$$\begin{aligned} x_b &= x_{C.G.} - x_{skid,aft} \\ y_b &= y_{skid} - y_{C.G.} \\ z_b &= z_{C.G.} \end{aligned} \tag{4.6.1}$$

Aft left skid extremity

$$\begin{aligned} x_b &= x_{C.G.} - x_{skid,aft} \\ y_b &= -y_{skid} - y_{C.G.} \\ z_b &= z_{C.G.} \end{aligned} \tag{4.6.2}$$

Forward right skid extremity

$$\begin{aligned} x_b &= x_{C.G.} - x_{skid,fwd} \\ y_b &= y_{skid} - y_{cg} \\ z_b &= z_{C.G.} \end{aligned} \tag{4.6.3}$$

Forward left skid extremity

$$\begin{aligned} x_b &= x_{C.G.} - x_{skid,fwd} \\ y_b &= -y_{skid} - y_{C.G.} \\ z_b &= z_{C.G.} \end{aligned} \tag{4.6.4}$$

The coordinates of each skid extremity $[x \quad y \quad z]_b$ correspond to the distances between the skids' extremities and the center of gravity *C.G.* in the body-axis system and are shown in figure 4.4. For example, in equation 4.6.1, the longitudinal distance x_b of the *aft-right* skid extremity is the difference between the longitudinal position (with respect to a reference line) of the center of gravity $x_{C.G.}$ and the position of the point on the skids $x_{skid,aft}$. This distance is negative as the aft of the skid is behind the center of gravity *C.G.* and the longitudinal axis x points forward. The coordinates found with equations

(4.6.1) to (4.6.4) will further be converted from the body-axis coordinate system into the modified Earth-axis coordinate system by use of the transition matrix $[T]_{cb}$ given in equations (4.2) and (4.3). The vertical distance between the helicopter's lowest point and the center of gravity, $z_{c,lowest}$, is found by computing the vertical distance z_c at each skid extremity, and then choosing the largest value. Equation (4.5) is further used to obtain the altitude at the lowest point Alt_{lowest_pt} since both the *C.G.* altitude Alt_{cg} and the distance $z_{e,lowest}$ are known. A logical flag (which is related to the altitude value) can be used to activate the ground dynamics model and to switch from a flight model based on stability and control derivatives to a simplified thrust and weight model, which is explained in the next section. This switch is necessary because the stability derivatives valid for flight are no longer valid when the helicopter touches the ground. A flight model based on stability and control derivatives estimates the forces and moments on the helicopter following small perturbation velocities from the trim condition. Upon touchdown, the helicopter is subjected to high linear and angular accelerations that cause high and rapidly changing perturbation velocities from the previous trim condition. If these high perturbation velocities are used in the stability and control matrices, then unreasonably large and non-valid forces and moments will be obtained which would lead to the non-convergence of the program.

4.3.4 Simplified Thrust and Weight Model

From the moment of touchdown to the end of the landing, the following assumptions are made for the aerodynamic, thrust and gravity forces and moments calculated in the body-axis coordinate system:

1. The aerodynamic moments acting on the rotor and the fuselage are negligible with respect to the moments resulting from the ground contact and the structural moments from the fuselage oscillations. Therefore, the moments from the stability derivatives are neglected and the aerodynamic moments from the controls derivatives are

progressively reduced as the main rotor thrust is decreased and the normal force between the ground and the helicopter increases.

2. The aerodynamic drag forces are negligible in comparison with the ground friction forces.

3. The resultant gravity and thrust forces in the x and y directions are zero.

Following these assumptions, the only significant resultant force F_z after touchdown is oriented in the z-direction and is calculated from the thrust T and gross weight $G.W.$ forces. This force is expressed in equation (4.7):

$$F_{z,flight} = G.W\cos\theta\cos\phi - T \tag{4.7}$$

In equation 4.7, the force given by the gross weight $G.W.$ is positive and the thrust force T is negative because in the body-axis system, the z-axis is positive when oriented downwards as shown in figure 4.5. The gravity forces act in the modified Earth axis coordinate system and so the additional term $\cos\theta\cos\phi$, multiplying the gross weight $G.W.$, appears.

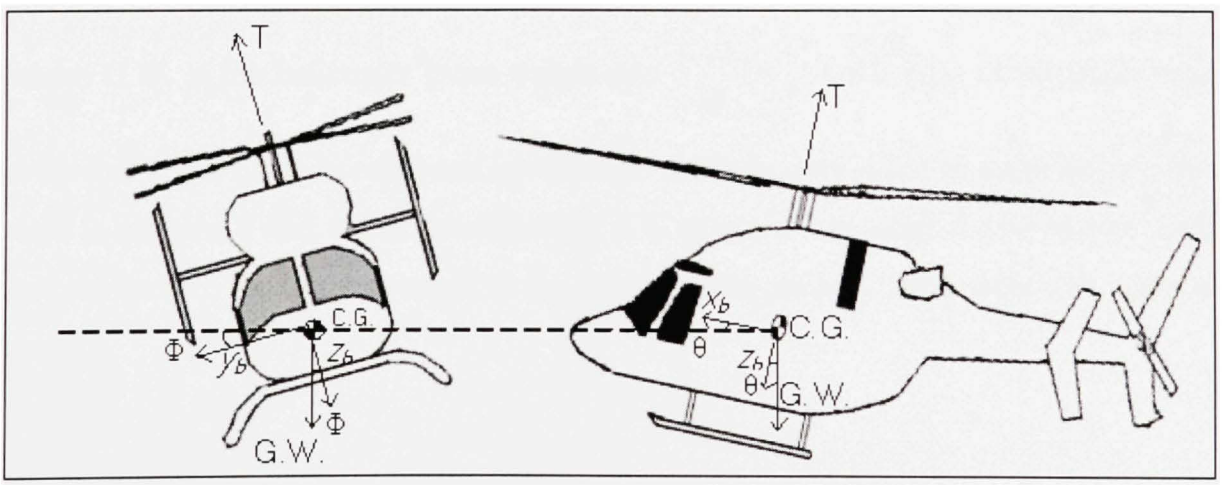

Figure 4.5 ***Free-body diagrams of the forces in the z-direction of the simplified flight dynamics model***

The helicopter thrust T is calculated from Prouty (1986) by use of equation (4.8):

$$T = \rho A C_T (\Omega R)^2 \tag{4.8}$$

where ρ is the air density, A is the main rotor disk area, Ω is the rotor blade rotational velocity and R is the radius of the main rotor. The term C_T is the thrust coefficient, a non-dimensional coefficient that is mainly a function of the collective position when the helicopter is on the ground. An expression to compute the helicopter's thrust coefficient as a function of the collective position when the helicopter is close to the ground (In Ground Effect) has been derived by use of hovering flight test data by Nadeau Beaulieu (2005). This expression is valid for normal and one-engine inoperative (the helicopter has two engines) landings when the collective position is close to the trim position. For autorotation, since the main rotor rotational speed Ω decays very quickly following touchdown, the thrust T of the helicopter is estimated by use of the following equation:

$$T = \left(\frac{\Omega_{auto}}{\Omega_{100\%}} \right)^2 G.W. \tag{4.9}$$

where $G.W.$ is the helicopter gross weight and $\left(\frac{\Omega_{auto}}{\Omega_{100\%}} \right)$ is the ratio of the main rotor rotational speed over its rotational speed when the helicopter is not in autorotation. This ratio is between 0 and 1. For this equation, it is assumed that when the helicopter lands in autorotation, the thrust is equal to the gross weight until the main rotor rpm starts to decay.

4.3.5 Normal force between the ground and the helicopter after touchdown

Following touchdown, the normal force between the ground and the helicopter is exerted in the z-direction in the modified Earth-axis coordinate system. When the helicopter touches down, the ground reaction is represented by the penalty method where a spring that pushes the helicopter upwards when the helicopter has the tendency to descend below the ground, and so we can represent the helicopter's vertical motion by the equation-of-motion of a spring-mass system. Recall that the general equation of a spring-mass system in one dimension is:

$$M\ddot{z} + C_z\dot{z} + K_z\Delta z = F_{z,aero} + F_{z,gravity} + F_{z,Thrust} \tag{4.10}$$

The terms on the right of equation (4.10) are the aerodynamic, thrust and gravity forces in the z-direction. Then, equation (4.10) may be written as follows:

$$\sum F_{z,\text{resultant}} = M\ddot{z} = F_{z,Aero} + F_{z,Gravity} + F_{z,Thrust} - C_z\dot{z} - K_z\Delta z \tag{4.11}$$

where $\sum F_{z,\text{resultant}}$ is the total force acting on the helicopter in the z-direction. With a three-dimensional body such as a helicopter, the vertical acceleration $\ddot{z}$ includes the coupling terms for a three dimensional body. The effects of the coupling terms will be taken into account in the rigid body 6 degree-of-freedom equations later in the helicopter's simulation. From equation (4.11), since the vertical forces calculated from aerodynamics, gravity and thrust are known, the vertical force equation for the ground dynamics model reduces to:

$$F_{z,GD} = -C_z\dot{z} - K_z\Delta z \tag{4.12}$$

This model assumes that the skid deflections of a flexible helicopter into a rigid ground is equivalent to the penetration of a rigid helicopter into the ground, so that the

deflection Δz after touchdown is equal to the negative of the altitude of the helicopter's lowest point $-Alt_{\text{lowest\_pt}}$. For example, if the altitude of the lowest point on the helicopter skids as computed by equation (4.5) is minus one inch, it means that the actual altitude of the helicopter's lowest point is 0 and the skid deflection is 1 inch.

In equation (4.12), K_z and C_z are linear functions of the skids' deflection as shown in the following equations:

$$K_z = K_1 K_\theta \Delta z$$
$$C_z = K_2 K_\theta \Delta z \tag{4.13}$$

where K_i are constants optimized to match the landing data. Multiplying the stiffness and damping by Δz insures that the vertical forces are applied progressively on the helicopter skids as they deform. The variable K_θ is a correction factor with respect to the pitch angle of the helicopter. This correction factor is necessary because when the pitch angle of the helicopter is high and only the rear part of the skid touches the ground (see figure 4.8), the skids deflect more than when the helicopter is low and the skids touch the ground completely (see figure 4.9). The evolution of the K_θ correction factor is represented in figure 4.6:

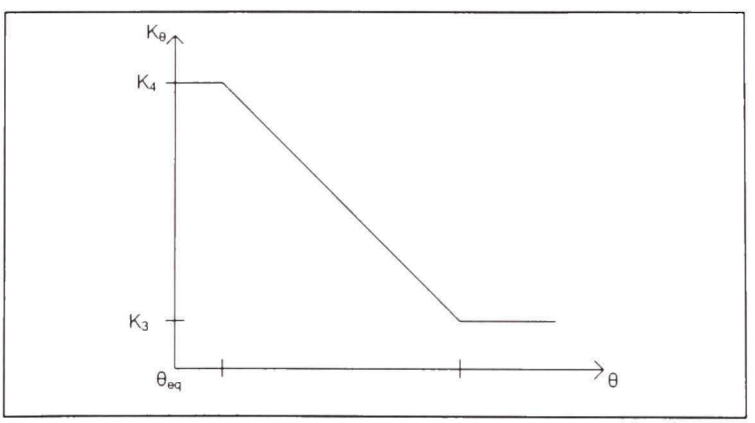

Figure 4.6 : ***Value of the K_θ correction factor as a function of the pitch angle θ. The parameter θ_{eq} is the equilibrium pitch angle of the helicopter.***

In the above figure, the term θ_{eq} is the pitch angle of the helicopter when is at equilibrium on the ground. This pitch angle is close to zero. Note that the correction factor K_θ would have a similar evolution if the initial pitch angle was negative.

4.3.6 Friction forces

The friction forces are produced by the friction between the ground and the helicopter's skids, and they decompose in the x and y directions. The magnitudes of the friction forces are expressed in the following equation:

$$F_{friction} = -\nu F_{z,GD} \tag{4.14}$$

This equation is only applied when the helicopter touches the ground. The friction force direction is opposite to the resultant tangential velocity direction as illustrated in figure 4.7.

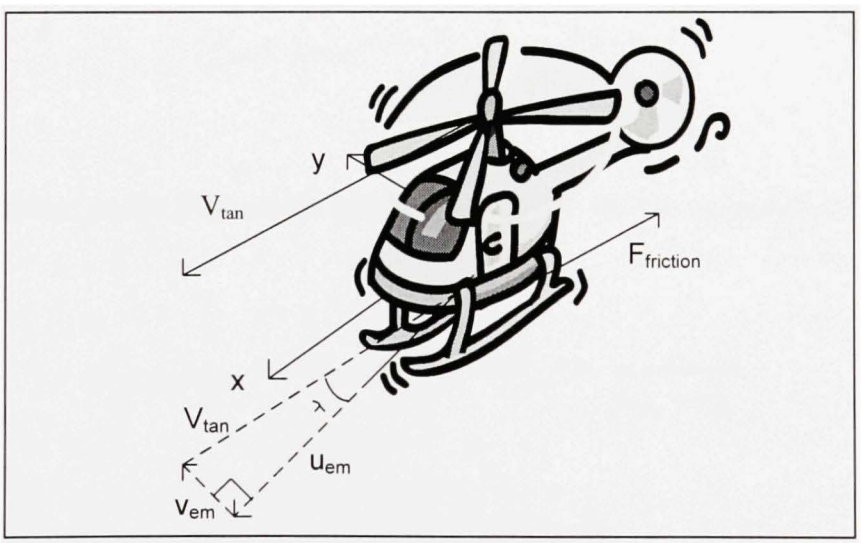

Figure 4.7 ***Friction force $F_{friction}$ opposite to the resultant tangential velocity at touchdown***

The forces acting in the x-directions can be expressed in figure 4.7 using the following equation:

$$F_{x,GD\_friction} = -F_{friction}\cos\lambda = -\nu' F_{z,GD}\cos\lambda \tag{4.15}$$

where the angle λ is defined in figure 4.7 and calculated with following equation:

$$\cos\lambda = \left(\frac{u_{cm}}{V'_{\tan}}\right) \quad \text{and} \quad \sin\lambda = \left(\frac{v_{cm}}{V'_{\tan}}\right) \tag{4.16}$$

where

$$V'_{\tan} = \sqrt{u_{cm}^2 + v_{cm}^2} \tag{4.17}$$

Therefore, we replace equations (4.14), (4.16) and (4.17) into equation (4.15), and the following expression for the friction force in the x-direction is obtained:

$$F_{x,GD\_friction} = -\nu' F_{z,GD}\left(\frac{u_{cm}}{\sqrt{u_{cm}^2 + v_{cm}^2}}\right) \tag{4.18}$$

The friction forces in the y-direction are found with the following equation:

$$F_{y,GD\_friction} = -F_{z,GD}\sin\lambda = -\nu F_{z,GD}\left(\frac{v_{cm}}{\sqrt{u_{cm}^2 + v_{cm}^2}}\right) \tag{4.19}$$

where

$$\nu = F_K + (F_S - F_K)e^{-\beta V'_{Tan}} \tag{4.20}$$

The friction coefficient ν increases with the decreasing helicopter velocity V_{Tan} given by equation (4.17) and is expressed as suggested by Sareen (1998) by equation (4.20). The friction coefficient ν approaches the static friction coefficient F_S when the helicopter is at rest. Equation (4.20) for the friction coefficient ν is validated for a number of 7 autorotation landing cases. This validation is realized by comparing the time histories of the ground tangential velocity V_{tan} of the helicopter model with the time histories of the ground tangential velocity of the landing records.

The validation studies give the values of static friction coefficient F_S equal to 0.4 and are situated within the range corresponding to concrete and steel (0.30-0.70) found in the literature from Beer and Johnston (2003). The kinetic friction coefficient F_K and the exponential decay coefficient β at touchdown are dependent upon the magnitudes of the helicopter's oscillations in roll and in pitch at the time of the touchdown. The parameter A, roughly proportional to the amplitude of these oscillations at touchdown, is defined from landing data as follows:

$$A = K_3\dot{\phi}^2 + K_4\dot{\theta}^2 + K_5\left(\phi - \phi_{eq}\right) + K_6\left(\theta - \theta_{eq}\right) \tag{4.21}$$

where ϕ_{eq} and θ_{eq} are the roll and pitch angles at rest on the ground (at the end of the landing record) and the coefficients K_i with i= [3, 4, 5, 6] are constants found by optimization to match the autorotation landing tests. This parameter A is computed at the time of the touchdown. The rate of change of the angles must be taken into account because if, at the time of touchdown, the helicopter has low roll or pitch angle, but a high roll or pitch rate, it means that it oscillates with high amplitude and it is necessary for the simulation to evaluate the angular rate in order to detect it.

The relationship between the kinetic friction coefficient F_K in equation (4.20) and the parameter A given by equation (4.21) can be expressed by the following equations:

$$F_K = K_7 + K_8 e^{K_9(A-K_{10})} \; ; \quad A \le K_{10} \qquad (4.22)$$

$$F_K = K_7 + K_8 \qquad ; \quad A > K_{10}$$

where the constants K_i with i = [7, 8, 9, 10] are optimized to match the landing data for 7 autorotation cases. Equation (4.22) indicates that if the helicopter does not oscillate in roll or pitch at touchdown, the term F_K is low. If the initial oscillation amplitude is increased, then the term F_K increases up to a maximum value of $(K_7 + K_8)$. An exponential term insures a progressive transition between the cases with and without oscillations at touchdown.

It was found that the exponential decay coefficient β varied linearly with the kinetic friction coefficient F_K as shown in equation (4.23):

$$\beta = K_{11} F_K + K_{12} \qquad (4.23)$$

where the constants K_i with i= [11, 12] were found by optimization to match the autorotation landing data. If the helicopter velocity with respect to the ground is zero, the static friction force is applied. The static friction is equal and opposite to any other force applied on the helicopter unless this force is higher than the friction force defined with the static friction coefficient. In this case, the helicopter starts moving and the dynamic friction equations are applied.

4.3.7 Rolling and pitching moments from ground dynamics after touchdown

When the helicopter touches down, the lowest part of the helicopter skids touches the ground as shown in figure 4.8. At this moment, there is an offset between the contact

normal force $F_{z,GD}$ given by the equation (4.12), the friction force in the x-direction (see equation (4.18)) and the helicopter's center of gravity.

This offset acts as a lever arm on the normal contact force which creates a rolling and a pitching moment. These moments reduce the helicopter angles until the skids completely touch the ground and are called « *Pivot* » moments. When the helicopter skids are completely on the ground, the helicopter keeps oscillating depending on the type of ground impact as illustrated in figure 4.9. At this time, the moments provided by the ground dynamics model are determined by the skids' stiffness and damping and are referred to as « *Oscillation* » moments. A logical flag is used in the computer programs to progressively switch between « *Pivot* » moments and « *oscillation* » moments depending on the angles values.

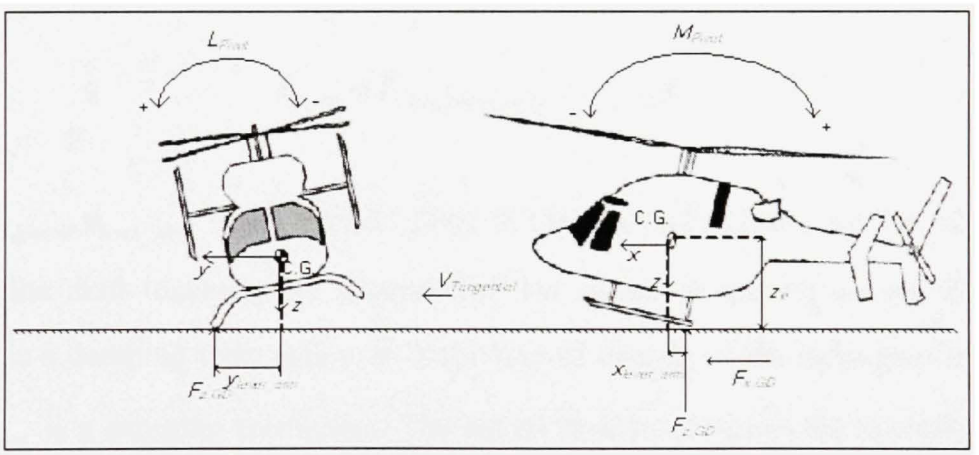

Figure 4.8 ***Rolling and pitching pivot moments calculated from the friction and initial ground contact forces at the helicopter touchdown point. These moments are called L_{Pivot} for rolling moment and M_{Pivot} for pitching moment.***

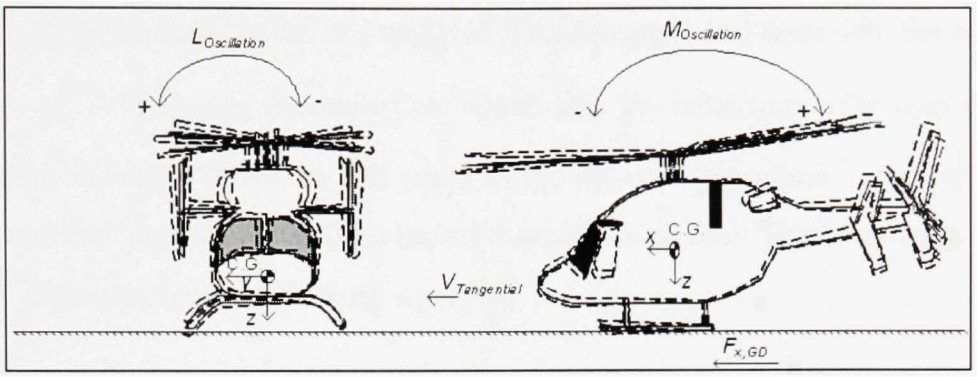

Figure 4.9 ***Helicopter roll and pitch oscillations following the skids' touchdown on the ground. The moments are called $L_{Oscillation}$ for rolling moment and $M_{Oscillation}$ for pitching moment.***

4.3.8 Rolling moments *L* during initial "Pivot" phase

The "*Pivot*" rolling moments L_{pivot} on the helicopter at touchdown can be represented by the following equation:

$$L_{pivot} = F_{z,ground} y_{lever\_arm} - C_{L,pivot}\dot{\phi} \tag{4.24}$$

where $F_{z,ground} y_{lever\_arm}$ is the vertical force in the z-direction times the lateral distance between the skid touching the ground and the center of gravity of the helicopter, $C_{L,pivot}\dot{\phi}$ is a damping term where $\dot{\phi}$ is the rate of change of the helicopter roll angle, and $C_{L,pivot}$ is a damping coefficient. The lateral distance between the touchdown point and the center of gravity y_{lever_arm} is defined using equations (4.25):

$$y_{lever\_arm} = (y_{Skid} + y_{cg})\cos\phi \qquad \phi > \phi_{eq} \tag{4.25.1}$$

$$y_{lever\_arm} = -(y_{Skid} - y_{cg})\cos\phi \qquad \phi < \phi_{eq} \tag{4.25.2}$$

The angle ϕ_{eq} is the equilibrium roll angle of the helicopter, and therefore, the sign of the term y_{lever_arm} will change depending on which side the helicopter rolls with respect to the equilibrium angle. When the roll angle of the helicopter is equal to the roll angle at equilibrium, the "oscillation" rolling moment equation is used. The damping coefficient $C_{L,pivot}$ is described by the following equation:

$$C_{L,pivot} = K_{13} + K_{14}\dot{\phi}^2 \tag{4.26}$$

The constants K_i with i= [13, 14] are empirical constants adjusted so that the model matches the landing data. Note that the roll-damping coefficient is limited to a maximum value.

4.3.9 Pitching moment during initial "Pivot" phase

The *Pivot* pitching moment M_{Pivot} can be determined by use of equation (4.27):

$$M_{Pivot} = -K_{15}F_{z,GD}x_{lever\_arm} + K_{16}F_{x,GD}z_e - C_{M,pivot}\dot{\theta} \tag{4.27}$$

where the term $-F_{z,ground}x_{lever\_arm}$ represents the product between the force in the z-direction and the longitudinal distance x_{lever_arm} between the touchdown point and the helicopter center of gravity. This term is negative because, as shown in figure 4.8, a negative contact force $F_{z,GD}$ (pointing up) in a negative x-coordinate (aft of the c.g.) will give a negative (nose down) pitching moment M_{Pivot}.

Note that the distance x_{lever_arm} is the same as the longitudinal position of the skid extremity in modified earth axes coordinates system.

The term $F_{x,GD} z_e$ in equation (4.27) represents the product between the friction force in the x-direction and the vertical distance between the touchdown point and the center of gravity. This term is positive because a friction force will point aft of the c.g. and give a negative value of the force $F_{x,GD}$ (defined as positive when the force points forward of the c.g.). A negative force $F_{x,GD}$ will cause a negative (nose-down) pitching moment M_{Pivot}.

The third term, $-C_{M,pivot}\dot{\theta}$, is a damping term where $C_{M,pivot}$ is the damping coefficient and $\dot{\theta}$ is the rate of change of the helicopter pitch angle. This term is negative because it opposes the helicopter angular velocity. It was found that the damping coefficient which best fits the landing data is proportional to the square of the rate of change in pitch angle, which can be expressed by the following equation.

$$C_{M,pivot} = K_{17} + K_{18}\dot{\theta}^2 \tag{4.28}$$

The constants K_i with i = [17, 18] are empirical constants adjusted so that the model matches the landing data. Note that the pitch-damping coefficient is limited to a maximum value.

If the helicopter touches down with a positive pitch angle and a positive pitch rate, the touchdown point penetrates into the ground much faster for the same rate of descent than if the helicopter had no positive pitch rate. When this situation occurs, a large negative moment is applied for a very short time between the ground and the helicopter and the helicopter starts pitching down much faster than if it had landed with no positive pitch rate. The pitching moment equation from the ground dynamics model requires an additional term to capture this effect properly. Equation (4.27) therefore becomes:

$$M_{Pivot} = -K_{15}\left(1 + K_{pitchrate}\right)F_{z,GD}x_{lever\_arm} + K_{16}F_{x,GD}z_c - C_{M,pivot}\dot{\theta} \qquad (4.29)$$

where the term $K_{pitchrate}$ in equation (4.29) is an additional correction factor added to the pitching moment equation at touchdown. This correction factor is necessary to model the large magnitude short duration moment that occurs when the helicopter lands with a positive pitch angle and a positive pitch rate. The term $K_{pitchrate}$ is represented by the following equation:

$$K_{pitchrate} = K_{19}K_{\theta,pitch}q_0 \qquad (4.30)$$

where q_0 is the initial positive pitch rate at touchdown and $K_{\theta,pitch}$ is a correction factor that progressively reduces the value of $K_{pitchrate}$ to zero as the pitch angle starts to decrease after touchdown. Equation (4.30) implies that the additional pitching moment at the instant of touchdown is proportional to the initial positive pitch rate. The value of the correction factor $K_{\theta,pitch}$ is represented in figure 4.10 where the term θ/θ_0 is the ratio of the helicopter pitch angle over its initial pitch angle at the instant of touchdown.

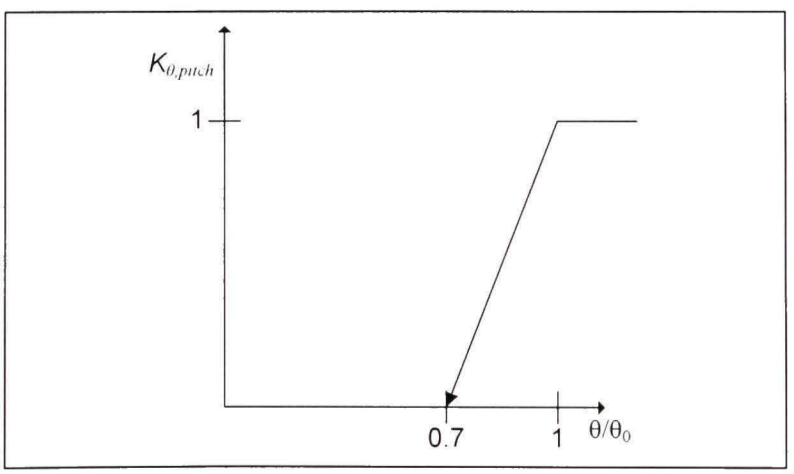

Figure 4.10 ***Value of the correction factor $K_{\theta,pitch}$ with respect to the ratio between the helicopter pitch angle and the pitch angle at touchdown.***

The term $K_{\theta,pitch}$ represented in figure 4.10 insures that the correction factor $K_{pitchrate}$ of equation (4.29) changes the pitching moment only for a brief time and that this effect decreases as the helicopter starts rotating toward its equilibrium pitch angle.

4.3.10 Rolling and pitching moments during the oscillation phase

Once the helicopter skids completely touch the ground, the helicopter oscillates around its equilibrium pitch and roll angles θ_{eq} and ϕ_{eq} as shown in figure 4.9. The moments due to these oscillations are found by use of equations:

$$L_{Oscillation} = -K_L\left(\phi - \phi_{eq}\right) - C_L\left(\dot{\phi} - \dot{\phi}_{eq}\right) \tag{4.31}$$

$$M_{Oscillation} = -K_M\left(\theta - \theta_{eq}\right) - C_M\left(\dot{\theta} - \dot{\theta}_{eq}\right) \tag{4.32}$$

We emphasize here that in these equations, no specific stiffness or damping components are represented, but the overall rolling and pitching motion resistance is represented in these equations when the helicopter is on the ground.

4.3.11 Roll stiffness and damping

The roll stiffness was adjusted to match the frequency of the roll oscillations after touchdown. The autorotation cases were used to adjust the stiffness because these landings contain more roll oscillations. This study determined that the roll stiffness that insured a good match with the landing data could be represented by a constant. The value of this constant doubles when its skids are completely on the ground after the initial rotation of the helicopter.

From autorotation landing cases, it was found that the roll oscillations' damped frequency and amplitude decay, which both depend on the roll damping C_L, vary with the helicopter's tangential velocity. The roll damping increases with a decreasing tangential velocity to fit the landing autorotation data. The roll damping C_L is modeled by use of equations (4.33):

$$C_L = K_{20} + K_{21} e^{-K_{22} V_{Tan}} \tag{4.33}$$

where the constants K_i with i = [20, 22] are adjusted to match the landing test data. From equations (33), is noticed the lower limit of the damping coefficient C_L is K_{20} when the tangential velocity is high and its upper limit is ($K_{20} + K_{21}$) when the tangential velocity is zero.

The damping coefficient C_L may be lower at higher tangential speeds due to the fact that as the helicopter oscillates in roll, the friction force increases alternatively on each skid, which produces a certain transfer of energy that excites the helicopter's rolling motion. The value of the constant K_{20} is lower if the helicopter has a high positive pitch angle (using the "pivot" pitching moment equation) than if it sits completely on its skids (using the "oscillation" pitching moment equation).

4.3.12 Relationship between rolling motion and sideward acceleration of the helicopter

After touchdown, since the roll center of the helicopter is not necessarily at the same position as the center of gravity, when the helicopter oscillates in roll, the pilot feels a sideward acceleration. This acceleration is reproduced in the model by an oscillating sideward force in phase with the helicopter roll angle oscillation, and is represented by the following equation:

$$F_{y,GD,oscillation} = K_{23} L_{Oscillation} \tag{4.34}$$

where K_{23} is a constant adjusted to match the lateral acceleration in the data. This force is added to the friction force in the y-direction from equation (4.19). There was no such relationship observed with the force in the x-direction and, in this direction, the force only comes from the friction with the ground. Therefore, the total force in the x and y-directions becomes:

$$\begin{aligned} F_{x,GD} &= F_{x,GD\_friction} \\ F_{y,GD} &= F_{y,GD\_friction} + K_{23} L_{Oscillation} \end{aligned} \tag{4.35}$$

This effect is most noticeable in autorotation landing cases where the helicopter roll oscillation is high.

4.3.13 Pitch stiffness and damping

4.3.13.1 Influence of the pitch angle θ on the pitch stiffness

From the flight test landing data, we concluded that the pitch stiffness is a non-linear function of the pitch angle θ as shown in figure 4.11:

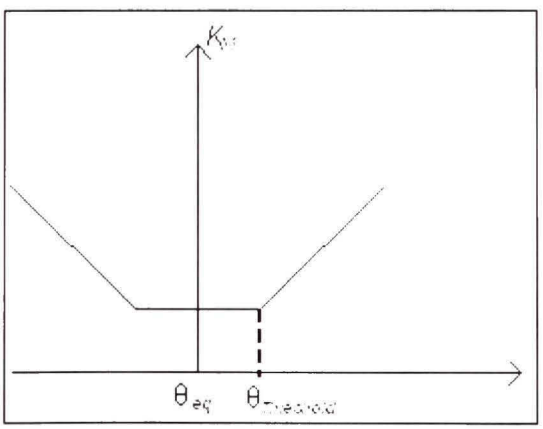

Figure 4.11 ***Variation of pitch stiffness K_M with the helicopter pitch angle θ***

In figure 4.11, it is known that if the pitch angle is above $\theta_{Threshold}$, the pitch stiffness increases with the pitch angle. Equations (4.36.1) and (4.36.2) were selected because in the data, the oscillation pitch angle θ never exceeds $\theta_{\text{threshold}}$ regardless of the landing test type.

$$K_{M,1} = K_{24}; \left(|\theta| < |\theta|_{Threshold} \right) \tag{4.36.1}$$

$$K_{M,2} = K_{24} + K_{25} |\theta - \theta_{Threshold}|; \left(|\theta| \geq |\theta|_{Threshold} \right) \tag{4.36.2}$$

where the constants K_i with i= [24, 25] are empirical constants found by optimization to match the landing test data.

4.3.13.2 Influence of the angular velocity $\dot{\theta}$ on the pitch damping

It was found from landing test data that the helicopter's pitch oscillations decayed very quickly, but the helicopter kept oscillating at low amplitudes and low angular velocities until its forward velocity on the ground was zero. A very good method to model these steady oscillations is to increase the helicopter pitch damping coefficient with the square of the pitch angular velocity $\dot{\theta}$; therefore, when the square of the angular pitch velocity θ is low, the damping is low and the helicopter has a steady oscillation, which is expressed with the equation (4.37):

$$C_M = K_{26} + K_{27}\dot{\theta}^2 + f\left(V_{Tan}\right) \tag{4.37}$$

where the constants K_i with i= [26, 27] are empirical constants found after optimization to match the landing test data. The rate of change of the pitch angle is squared because

this formulation gave a better match with the landing data. The term $f(V_{Tan})$ is a function of the forward velocity, which will be explained in the following section.

4.3.13.3 Influence of the forward velocity

As noted above, it was observed that pitch oscillations last until the forward velocity of the helicopter is zero. This means that the helicopter oscillates longer (6 to 20 seconds) during autorotation landings when the forward velocity at touchdown is high than for one-engine landings (less than 2 seconds) when the forward velocity at touchdown is very low.

These situations were modeled by a damping that increases exponentially with decreasing ground tangential velocity V_{Tan} (forward velocity). An inverse exponential dependence of the pitch damping C_M on the helicopter tangential velocity V_{Tan} models this effect very well and so equation (4.37) becomes:

$$C_M = K_{26} + K_{27}\dot{\theta}^2 + K_{28}e^{-K_{29}V_{Tan}} \tag{4.38}$$

The exponential decay term $K_{30}e^{-K_{31}V_{Tan}}$ is chosen to obtain an additional damping of zero at high velocity, and a constant additional damping at low velocity. A general expression for pitch stiffness K_M and pitch damping C_M can be obtained by a combination of equations (4.36), (4.37) and (4.38):

$$K_{M,1} = K_{24} \qquad \left(|\theta| < |\theta|_{Threshold}\right) \tag{4.39.1}$$

$$K_{M,2} = K_{24} + K_{25}\left|\theta - \theta_{Threshold}\right| \qquad \left(|\theta| \geq |\theta|_{Threshold}\right) \tag{4.39.2}$$

$$C_M = K_{26} + K_{27}\dot{\theta}^2 + K_{28}e^{-K_{29}V_{Tan}} \tag{4.39.3}$$

4.3.14 Yawing moment from yaw damping term $N_{GD\_damping}$

When the helicopter touches down, the only yawing moment N from the ground is a damping term C_N due to friction, which reduces the heading angle rate of change of the helicopter $\dot{\psi}$ following touchdown. The yawing moment N can therefore be expressed by use of equation (4.40):

$$N_{GD\_damping} = -C_N \dot{\psi} \tag{4.40}$$

Contrary to the roll and pitch equations, there is no heading equilibrium position and therefore, there is no stiffness term. As for the roll and pitch damping terms, the damping is negative because it opposes the yaw motion $\dot{\psi}$. The general expression for the yaw damping varies with the normal force between the helicopter and the ground and the helicopter pitch angle. There are two different expressions for the yaw damping coefficients:

$$\begin{aligned} C_{N,pivot} &= -K_{30} F_{z,GD} \\ C_{N,oscillations} &= -K_{31} F_{z,GD} \end{aligned} \tag{4.41}$$

where $C_{N,pivot}$ is the yaw damping when the helicopter lands with a positive pitch angle and rotates with only the rear part of its skids touching the ground (see figure 4.8), and $C_{N,oscillation}$ is the yaw damping when the skids are completely on the ground and the helicopter pitch angle is small (see figure 4.9). It is important to remember that the normal force $F_{z,GD}$ is defined negative when in upward direction, therefore, after touchdown, a normal force from the ground pointing up gives a negative value of $F_{z,GD}$. The slopes K_{33} and K_{34} are very high at the beginning of the touchdown and are lower when the weight on the skids is more than a certain threshold. This is illustrated in figure 4.12:

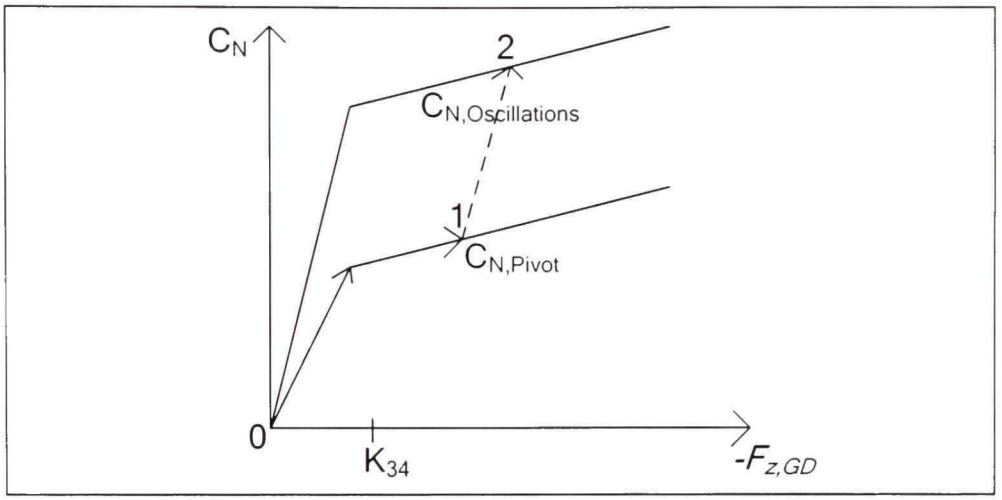

Figure 4.12 ***Variation of the damping terms*** $C_{N,pivot}$ ***and*** $C_{N,oscillation}$ ***vs. the negative value of the normal force between the ground and the helicopter (defined as negative up).***

The arrows and the dotted line on figure 4.12 show a typical evolution of the yaw damping coefficient during a landing. Point 0 represents the initial contact with the ground when there is no yaw damping because there is no force between the ground and the helicopter. From point 0 to point 1, the helicopter pitch angle is positive, meaning that only the rear part of the landing gear touches the ground and the $C_{N,pivot}$ damping curve is used. Regardless of the landing record, the damping increases rapidly to point 1 (within a fraction of a second) as more weight is added to the skids. The position of point 1 will be different for each landing record depending on the normal force between the ground and the helicopter, which, in turn, depends on the helicopter thrust and weight. The yaw damping will be nearly constant while the helicopter rotates to its equilibrium pitch angle.

From point 1 to point 2, the helicopter pitch angle goes progressively to the equilibrium pitch angle θ_{eq}, and the damping coefficient is progressively changed to $C_{N,oscillations}$. The final position of point 3 will also be different for each landing record. Note that the

normal force between the ground and the helicopter is almost always higher than K_{35} which is the intersection between the two slopes. It would have been possible to match the data using only the slopes when the normal force is higher than K_{33} and defining an initial value of the yaw damping above zero, but it would not have been physically correct because the yaw damping coefficient must be zero at the initial contact between the helicopter and the ground.

4.3.15 Yawing moment from roll coupling N_{GD_roll}

The second component of the yawing moment is from a coupling between the roll and yaw motion of the helicopter. This coupling produces oscillations in yaw that are in phase with the oscillations in roll. This is calculated by use of following equation:

$$N_{GD\_roll} = K_{33} L_{Oscillation} \tag{4.42}$$

This coupling is only present after the helicopter has done its roll rotation and is oscillating in roll. Using the damping and coupling terms, the yawing moment on the helicopter from the ground dynamics model is described by use of equation (4.43):

$$N_{GD} = N_{GD\_damping} + N_{GD\_roll} \tag{4.43}$$

4.4 Results

The ground dynamics model has been validated for the following 14 landing cases, but results obtained with this model were presented for 8 landing cases:

- One engine inoperative (4 cases)
- Autorotation landing (4 cases)

Four cases of each category are shown in this paper. For the helicopter model validation, the initial conditions of the simulation were taken from the measured data.

For the ground dynamics model validation, we use the attitudes calculated from flight test data as initial conditions for the ground dynamics simulation. In figures 4.13 and 4.14, the model output is represented by a full line and the dotted line represents the FAA (Federal Aviation Administration) tolerance bands. The FAA tolerance bands are the values obtained from the landing data plus or minus the allowable errors of the simulation model as defined by the FAA regulations. For example, if the tolerance band is $\pm 1.5^{\circ}$, the upper dotted line represents the value from the landing data plus 1.5 degrees and the lower dotted line represents this value minus 1.5 degree. A simulation model is considered acceptable and certifiable by the FAA if it remains between these two dotted lines. The FAA tolerances bands are:

- One engine inoperative landings: $\phi : \pm 1.5^{\circ}; \theta : \pm 1.5^{\circ}; \psi : \pm 2^{\circ}$ and the tangential velocity $V_{Tan} = \pm$ 3 knots.

- Autorotation landings: $\phi : \pm 2^{\circ}; \theta : \pm 2^{\circ}; \psi : \pm 5^{\circ}$ and there is no FAA tolerance band on tangential velocity V_{Tan}.

4.4.1 Results for One-Engine Inoperative OEI cases

Figure 4.13 shows the time histories of the roll angles ϕ, pitch angles θ, yaw angles ψ and velocity V_{Tan} for the 4 OEI (One Engine Inoperative) landing cases. The other landing cases are quite similar in form and for this reason, are not shown here. Please note that no numbers are shown on the x and y axes for proprietary reasons. From figure 4.13, it is clear that the ground dynamics model outputs are within the FAA tolerance bands. In these cases, the simplified thrust and weight model was applied just before

touchdown, but when the helicopter touches down, then the « *pivot* » rolling L and the pitching M moments (given by equations (4.24) and (4.29)) return the helicopter slowly to its equilibrium position. The equations for the « *oscillation* » moments L and M are also used (equations (4.31) and (4.32)). It can also be observed that the tangential velocity V_{Tan} decreases slowly to zero due to the friction. In the other cases, the tangential velocity was already zero and remained at this value. The rate of change of the yaw angle slowly decays due to yaw damping. Notice that for a simulator qualification, it is required to demonstrate the simulator that the simulator match the landing data for three OEI landing cases: one category A landing, one category B landing and one landing following a rejected take-off. In the ground dynamics model, a good match was achieved for seven OEI landing cases, which is more than sufficient to qualify it the model for a level D flight simulator.

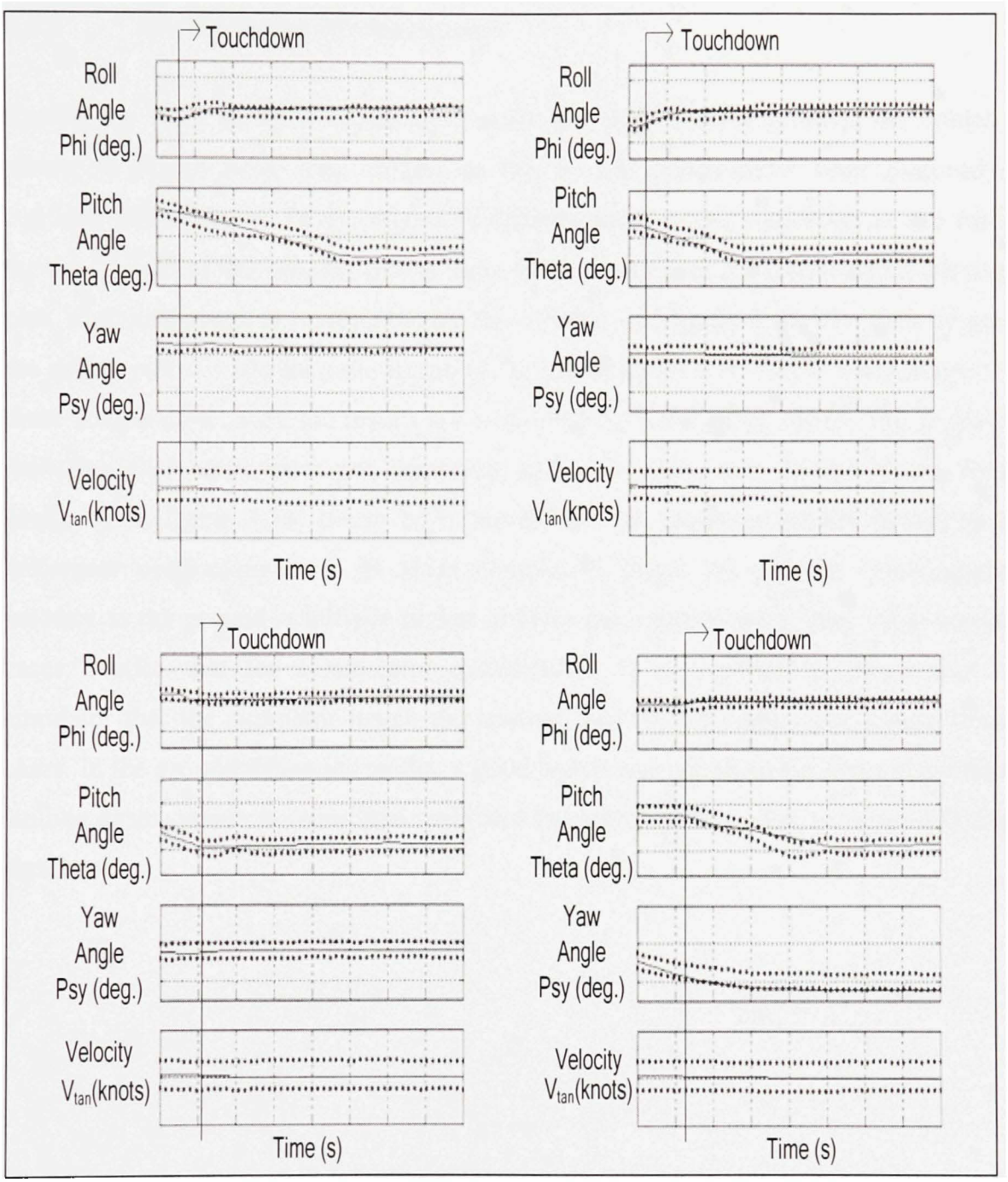

Figure 4.13 ***Results for 4 OEI landing cases***

4.4.2 Results for autorotation cases

Results for 4 out of the 7 autorotation cases that were used to validate the model are shown in figure 4.14. The values on the x- and y-axis have been removed for confidentiality reasons. In this figure, in order to improve the readability of the results, the time length of the velocity plot is three times longer than the timescale on the angles plot. The time length is represented by the variable x in figure 4.14. The time length of the angles plot is x and the time length of the velocity plot is $3x$ (three times longer). In these autorotation cases, the results are within the FAA tolerance bands. The results for the other five cases were not displayed, but they were very similar to the results displayed in figure 4.14. It can be observed that the model is mostly driven by the helicopter oscillations once its skids completely touch the ground. The tangential velocity to the ground is initially higher in these autorotation cases than for one-engine cases. Notice that for a simulator qualification, it is required to demonstrate the simulator that the simulator match the landing data for only one autorotation landing cases. In the ground dynamics model, a good match was achieved for seven autorotation landing cases, which is more than sufficient to qualify it the model for a level D flight simulator.

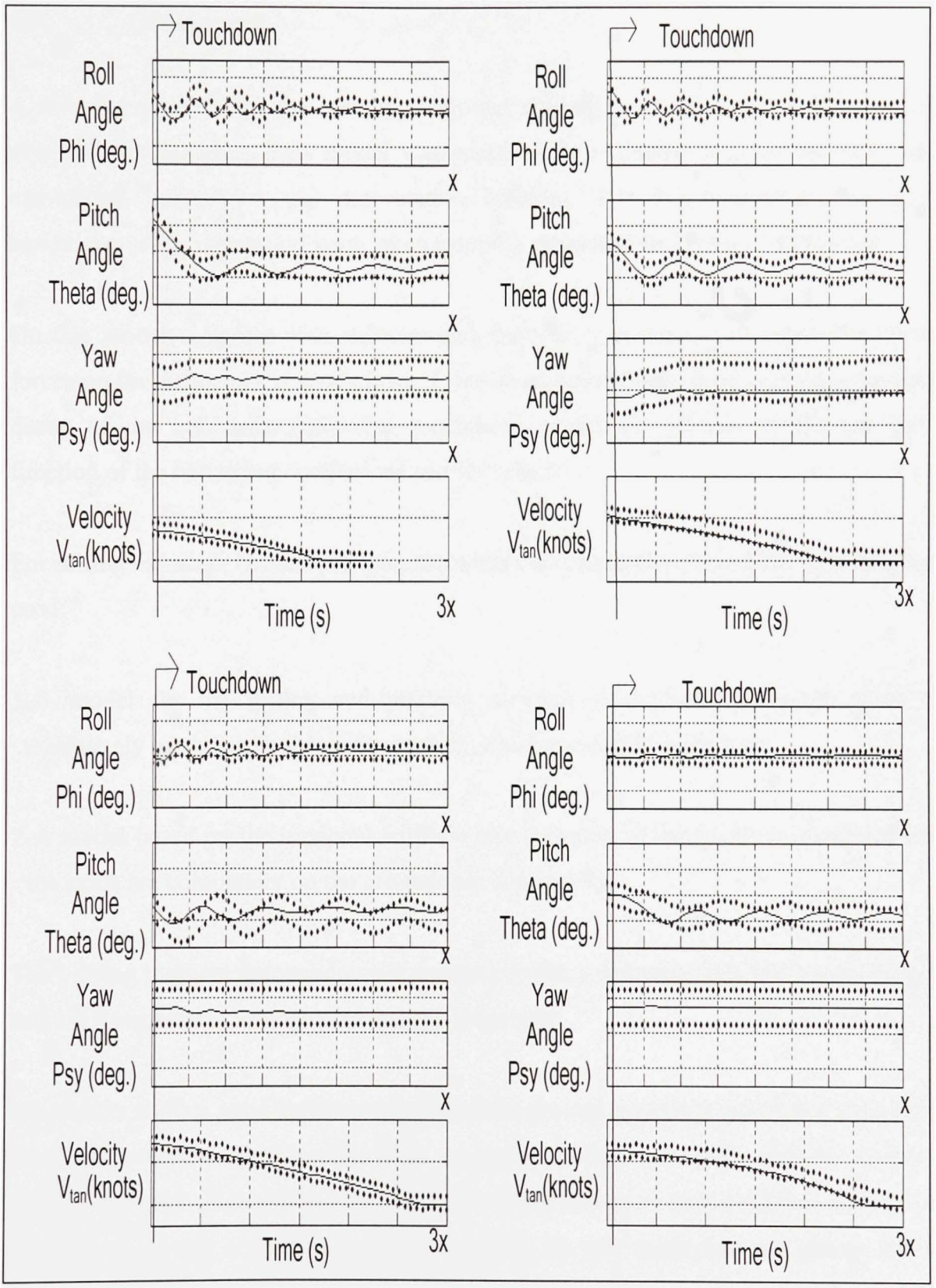

Figure 4.14 ***Results for four autorotation landing cases***

4.5 Conclusions

A new formulation for the helicopter ground dynamics was developed based on the available landing data. This model was successfully validated with landing data from one-engine inoperative and autorotation landings. For this validation, the initial conditions of the simulation were taken from the measured data before touchdown.

On this model, a spring with stiffness and damping was used to calculate the normal forces on the helicopter at touchdown. Friction equations were used to model the speed decay of the helicopter following touchdown while the friction coefficient was a function of the helicopter oscillations and its velocity.

For rolling moments (L) and pitching moments (M) calculation, two different models are used:

1-A model for the rolling and pitching moment is applied before the skids are completely on the ground (see figure 4.8), which rotates the helicopter.

2-A model based on the torsional stiffness and damping of the fuselage is applied once the skids are completely on the ground (see figure 4.9).

The yawing moment at touchdown is computed with a damping term that varies with the normal force between the ground and the helicopter.

The results show a good agreement between the ground dynamics model and the landing data. For the one-engine landing cases, as shown in figure 4.13, the absolute values of the roll and pitch angles of the helicopter decreases slowly to their equilibrium positions. The yaw angle rate of change also decrease until the yaw angle does not change and the tangential velocity decreases slowly to zero. In the autorotation cases (see figure 4.12),

the roll angle of the helicopter oscillates with larger amplitude than for one-engine cases and the oscillation decays to the equilibrium position. A smaller oscillation is also present in the pitching motion. The rate of change of the yaw angle and the tangential velocity decays to zero due to the ground friction.

4.6 References

Baraff, David, July 1989, "Analytical method for dynamic simulation of non-penetrating rigid bodies", Association for Computing Machinery's Special Interest Group on Graphics and Interactive Techniques, Volume 23, Issue 2, 1989, pp.223-232

Baraff, David, "Non-penetrating rigid body simulation", Association for Computing Machinery's Special Interest Group on Graphics and Interactive Techniques, 1989, pp.223-232.

Beer, Ferdinand P.P., E. Russel Johnston Jr., and Elliot R. Eisenberg, George H. Staab, 1999, *Vector Mechanics for Engineers*, Static and Dynamics, McGraw-Hill Science/Engineering/Math, 7th Edition, McGraw-Hill Science/Engineering/Math, 624 pages.

Federal Aviation Administration, 1994, *Helicopter Simulator Qualification*, Advisory Circular AC 120-63, Federal Aviation Administration, US department of transportation

Guendelman, Eran, Eric Bridson, Ronald Fedkiw, 2003, "Non-convex rigid bodies with stacking", Association for Computing Machinery's Special Interest Group on Graphics and Interactive Techniques, Volume 22 issue 3 , pp. 871-878.

Johnson, Eric N., Paul A. DiBietto, 1997, "Modeling and simulation for small autonomous helicopter development" in AIAA 1997 modeling and simulation technologies conferences, New Orleans, Aug. 11-13, pp.1-11.

Nadeau Beaulieu Michel, Andrei Vladimir Popov, Ruxandra Botez, Adrian Hiliuta, Ruxandra Popescu, Njuki Mureithi, (2005) *Methodology for Aerodynamic Model Generation for the B-427 Helicopter in Hover in ground effect,* Eleventh Australian International Aerospace Congress, Melbourne, Australia, March 13-17, 2005.

Nelson, Robert C., 1998, *Flight Stability and Automatic Control,* second edition, Mc Graw Hill, pp.102-103.

Prouty, Raymond W., 1986, *Helicopter Performance Stability and Control,* Malabar, Florida: Krieger Publishing Company, 731 p.

Sareen, Ashish K., Michael R. Smith, John V. Howard, 1998, "Helicopter skid gear dynamic drop analysis and test correlation", American Helicopter Society 54th Annual Forum, Washington D.C., May 20-22, pp.1267-1274

CHAPTER 5

INTRODUCTION TO THE SECOND PAPER

The second paper of this thesis deals with parameter estimation methods applied to the B-427 helicopter. This time, a *black box* model is used to simulate the main rotor torque, tail rotor torque, engine torque and main rotor speed. These parameters are calculated in time with state-space models which use non-linear inputs. The parameters describing these state-space models have been identified with the subspace system identification method described in Chapter 3. These parameters were further optimized with the Levenberg Marquardt minimization algorithm. The model was implemented as a simulation and as a prediction, and could be used in the following applications:

-A mathematical model of a flight simulator

-A health monitoring system on a helicopter

-A control system to limit these parameters below their maximum value during helicopters operation

Submitted to the following journal:

1-Nadeau Beaulieu M., Botez, R. M., 2007, *Prediction of main rotor, tail rotor and engine parameters from flight tests*, Journal of Aerospace Engineering.

Accepted to the following conference:

1-Nadeau Beaulieu M., Botez, R. M., 2008, *Simulation and prediction of the helicopter main rotor, tail rotor and engine parameters by using the subspace system identification method*, 49th AIAA Structure, structural dynamics and material conference, Schaumburg, Il., USA, 7-10 Apr. 2008

CHAPTER 6

SIMULATION AND PREDICTION OF THE HELICOPTER MAIN ROTOR, TAIL ROTOR AND ENGINE PARAMETERS BY USE OF SUBSPACE SYSTEM IDENTIFICATION METHOD

Nadeau Beaulieu M., Botez R. M.

Laboratoire avancé de recherche en commande, avionique et aéroservoélasticité, École de Technologie Supérieure, 1100, Notre-Dame West, Montréal, Québec, Canada, H3C 1K3

6.1 Abstract

In the framework of this research project, the main rotor torque, tail rotor torque, engine torque and main rotor speed of a helicopter in forward flight are estimated by using a state space model from flight tests data. The state space model inputs are non-linear terms made of combinations of pilot controls and helicopter states. The model simulates the helicopter outputs while knowing the states and controls at all times. It was also implemented as a prediction tool, for possible use in an envelope protection flight control system in which the states, controls and outputs are known at the present time, and predict the future helicopter states and controls following to pilot controls time history. The state space model parameters are identified by using the subspace identification method, a relatively recent non-iterative algorithm which constructs an observability matrix from input and output data and uses this matrix to obtain the state-space matrices. The obtained parameters are then optimized with the Levenberg-Marquardt output-error method. A comparison of the results with and without optimization is also conducted. The results show that the subspace method provides a good estimate of the outputs within the FAA tolerance bands and that these results can further be improved by use of the minimization algorithm. The generated model using

the subspace method is found to be very good for prediction applications, which makes it a promising model for flight control simulator applications.

6.2 Introduction

In order to ensure the safe and efficient operation of a rotorcraft, it is very important to understand the relationships between the parameters related to the aerodynamics and controls of rotors and engines, such as the main rotor torque, the pilot inputs and the helicopter states. The focus of this paper is the generation of mathematical model for the main rotor torque, tail rotor torque, engine torque and main rotor speed of a helicopter.

The model uses as inputs the rotorcraft states and the pilot controls. Different implementations of such a mathematical model can be used in different applications such as flight simulators and envelope protection control systems.

In a flight simulator application, the rotorcraft states are given by the flight simulator model, which uses these parameters in conjunction with the pilot control inputs to estimate the main rotor torque and displays its value in the cockpit. It is very important for the pilot safety to learn how his manoeuvres affect the rotors and the engines, because of the fact that when these parameters exceed certain limits, they can have a detrimental effect on the helicopter's fatigue life and its handling qualities safety.

Such a mathematical model can also be used in envelope protection control systems. In this case, it is necessary to ensure that some values remain within their prescribed limits. In this paper, the limit parameters are the model outputs as defined previously. In order to prevent such a limit violation, it is necessary to know the relationship between the future control inputs and the future values of the limit parameters. This relationship can be found by constructing a predictive model which uses the states and outputs current values from flight test data to predict the future values of the outputs based on assumed futures values of the pilot inputs. Inverting this model provides the future control inputs

that would result in a limit violation. Once these relationships are known, according to Jeram et al. (2005) there are two possible options: autonomous restraint, aural and visual limit cues and carefree manoeuvring control systems.

In the first option, the control system can override the pilot and change the control inputs in order to avoid reaching the limit, which is suitable whenever a violation of the limit can lead to a catastrophic accident. Examples of such limits are the rotorcraft stall and airspeed limits. The main advantage of this option is that the pilot workload is greatly reduced. However, such systems de-emphasize pilot judgement in critical decisions. In the second option, the system only warns the pilot of approaching limits without overriding his actions. This method is the opposite of the first one in the sense that the pilot has full authority over the aircraft, but a greater workload. Whenever the pilot needs to perform very aggressive manoeuvres, it may also lead him to be over cautious to avoid violating the limit, which leads to a decreased aircraft performance.

There is also a third option which is a compromise between the first two options. In this case, a progressive resistance is implemented on the helicopter controls as the limit is approached. The pilot does not need to monitor his controls to know how far the limit is because he knows it intuitively from the resistance exerted on the controls. If he wants to perform an aggressive manoeuvre, the pilot can chose to follow the force cue and allow it to guide him along the helicopter limit or he can chose to override it if he requires a greater vehicle performance regardless of the risk (for example, to avoid an obstacle). This option is suitable when a limit violation (e.g. main rotor torque or main rotor speed) can be detrimental, but not necessarily catastrophic. According to a survey of 70 UK military helicopter pilots done by Massey et al. (1988), 75 % of the pilots estimated that having to monitor the torque limit had a significant impact on mission performance, 60 % of them also believed the same true for rotor speed limits.

In this paper, a predictive model that could be used in such a control system is generated. The model structure used in this paper so as to estimate the limit parameters is a state space model with non-linear inputs. This state space model was constructed with B-427 helicopter flight test data. In order to obtain the torque response to a broad range of motions, 2311 manoeuvres described by Jategaonkar (2006) were performed on each helicopter control at different true airspeeds, altitudes, gross weights and center of gravity positions. The parameters in the state space matrices were identified by means of the subspace identification method. The MATLAB® implementation of the subspace system identification method explained in details by Ljung (1999, 2006) is an efficient non-iterative algorithm which uses inputs and outputs data to obtain directly the system observability matrix. This matrix is further used to obtain the A,B,C,D state space matrices. Because it is non-iterative, the subspace identification method is much faster than alternative methods which require optimization. Furthermore, it is not affected by iterations, such as the possible convergence of the solution toward the local minimum instead of the global minimum and does not require a-priori knowledge of the parameters values. Following the application of the subspace method to obtain the initial guesses for the parameter values in the state space matrices, an output error method based on the past Levenberg-Marquardt minimization algorithm was used to refine the results. The applications of these methods in the Aerospace Industry will be detailed next.

In 1995, Howitt used a simplified mathematical model to estimate the engine torque and main rotor torque of the BO-105 helicopter following a collective step input and designed carefree control laws based on this simplified model. Many of the research efforts that followed used Neural Networks to predict the limit parameters' future value. Menon et al. (1996) were able to predict the main rotor speed of a helicopter at one sample time instant in the future with an adaptive linear neural network model. In this algorithm, the neural networks' weights were adjusted on-line using the prediction error and showed one case with a low prediction error (less than 0.25 %). Neural networks

were also used to predict the main rotor hub moments for a load monitoring technology application on the SH-60 helicopter by McCool (1998). In 1999, Horn and Prasad used offline trained Neural Networks to predict the control inputs that would result in a violation of the torque, load factor or angle of attack limits on the V-22 aircraft in dynamic trim. The dynamic trim was defined as the condition for which the fast aircraft states (angular rates, etc.) have reached steady state and the slow states (Euler angles, TAS, etc.) continued to vary in time. However, they did not worry about the limited parameters' peak value following a control input. Yavrucuk (2001, 2002) also estimated the dynamic trim limited parameter of the rotorcraft, but used an approximate linear model corrected by an on-line trained adaptive neural network. Sahasrabudhe, Horn and Sahani (2002, 2004, 2005) also used a neural network to estimate the value of a limited parameter dynamic trim and added an approximate linear model to obtain its peak value.

In the methodology section provided below, we will explain the flight conditions covered by the model as well as the manoeuvres used in its identification. Then, the implementation of the model in the simulation and the details on the subspace identification algorithm will be given. The method to use the model as a prediction tool for an envelope protection control system application will further be presented.

6.3 Methodology

6.3.1 Flight conditions and manoeuvres used in the model identification and validation

Different state space models were identified for flight tests conditions expressed by different altitudes, gross weights and center of gravity positions in forward flight, see Table 6-1. For each flight test condition, a number of different forward flight records (see column 6 in Table 6-1) was used to identify the models. The other records were used to validate them. One record consists of a flight time history which starts when the helicopter is at trim, then the pilot performs a manoeuvre and records the effects of his manoeuvre on the helicopter states and controls. The manoeuvres are here called **2311**

because they consist of **4** step inputs lasting respectively **2, 3, 1** and **1** seconds. Such manoeuvres are used because they excite both the short and long period frequencies of the helicopter motion.

Table 6.1

Flight tests conditions used to identify and validate the proposed model in forward flight

Flight test condition	Gross Weight	Center of gravity position	Altitude Range	True airspeed range	Number of records used to identify the model	Number of records used to validate the model
	Light or Heavy	Forward, Mid or Aft	* 1000 ft	knots		
1	Heavy	Aft	4-8	60-160	42	12
2	Heavy	Aft	8-12	50-160	49	20
3	Light	Aft	3-6	40-110	28	12
4	Light	Aft	6-8	35-130	69	22
5	Light	Aft	8-10	50-70	32	12
6	Heavy	Mid	5-8	30-130	21	4
7	Heavy	Fwd	0-4.5	40-140	33	12
8	Heavy	Fwd	4.5-7	30-130	30	8
9	Heavy	Fwd	7-10	30-130	56	20
10	Light	Fwd	4-9	30-130	49	16
Total					409	138

For each record, a **2311** manoeuvre was performed on one of the helicopter's controls (collective, longitudinal cyclic, lateral cyclic and pedal).

Figure 6.1 shows an example of these **2311** manoeuvres:

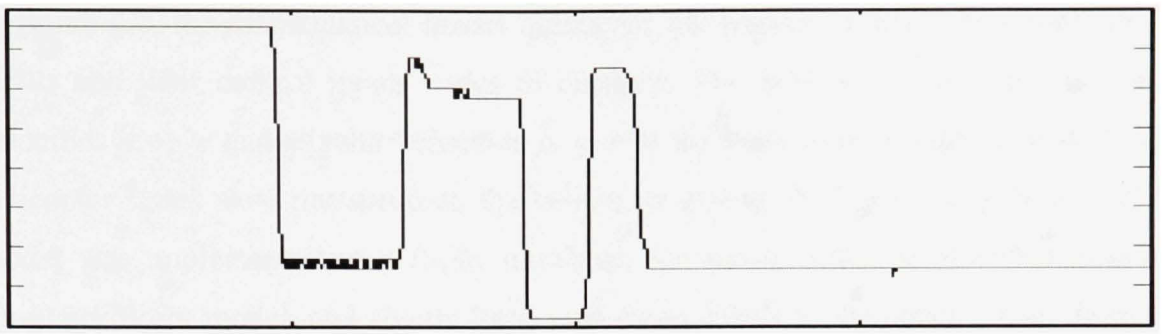

Figure 6.1 ***Sample time history of a control position during a 2311 manoeuvre***

In the following section, the model inputs and outputs will be described.

6.3.2 State Space model inputs and outputs

In this paper, the main rotor torque, tail rotor torque, engine torque and main rotor speed are modeled by using state-space models. Figure 6.2 shows the structure of the mathematical model for the estimation of these parameters:

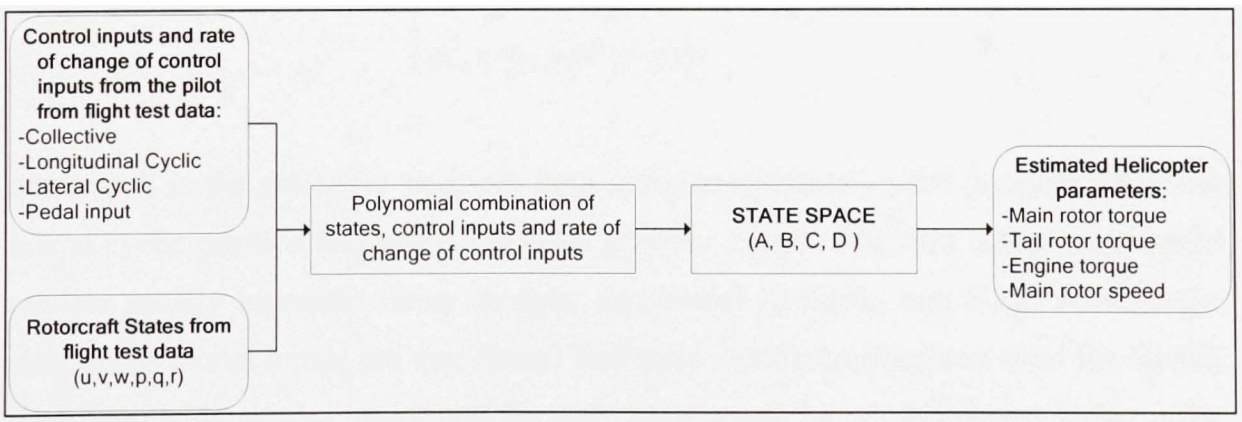

Figure 6.2 ***State space model architecture for the identification of the main rotor torque***

In figure 6.2, the mathematical model inputs are the helicopter states, the pilot control inputs and their control inputs' rates of changes. The helicopter states are the linear velocities u, v, w and angular velocities p, q, r in the body axis coordinate system. The helicopter states were measured on the helicopter during the flight test program. If the model was implemented in a flight simulator, the states could be provided from the simulator flight model, and should have very close values to the states values from the flight test data if the helicopter flight model would be properly designed. The actual state space model inputs are the previously mentioned inputs and higher order terms made of products of different inputs. For example, for the main rotor torque, the state space model inputs are:

$$Inputs_{MainRotorTorque} = \begin{bmatrix} coll, long, lat, ped, \frac{dcoll}{dt}, u, v, w, p, q, r, \\ coll^2, coll \cdot \frac{dcoll}{dt}, coll \cdot q, coll \cdot u, u \cdot \frac{dcoll}{dt}, \\ v \cdot \frac{dcoll}{dt}, q \cdot \frac{dcoll}{dt}, coll^2 \cdot \frac{dcoll}{dt}, coll^2 \cdot u, \\ w^2, r \cdot w, ped^2, r \cdot ped \end{bmatrix} \tag{6.1}$$

where *coll* is the collective position, *long* is the longitudinal cyclic position, *lat* is the lateral cyclic position and *ped* is the pedal position. Notice that even though state space models usually represent linear models, this model is highly non-linear because the state-space model inputs are non-linear. The same model structure was used for the tail rotor torque, the engine torque and the main rotor speed, but with different higher order terms. The higher order terms were selected by trial and error and each non-linear term was kept only if it improved the results for the records used for the identification and for the records set aside for the validation.

6.3.3 Subspace identification method

The State Space matrices parameters (see figure 6.2) were obtained by using the subspace system identification algorithm, which is briefly described in this section. Generally, a discrete linear model is defined by the following equations (6.2.1) and (6.2.2):

$$x(t+\Delta T)_{n\times 1} = \hat{x}(t+\Delta T)_{n\times 1} + K_{n\times o}v(t)_{o\times 1} \tag{6.2.1}$$

$$y(t)_{o\times 1} = \hat{y}(t)_{o\times 1} + v(t)_{o\times 1} \tag{6.2.2}$$

where t is the time and ΔT is the time increment equal to the record sampling rate. In equation (6.2.2), the vector $y(t)$ represents the system's *measured* outputs from flight test data, $\hat{y}(t)$ represents the system's *estimated* outputs and $v(t)$ represents the error between the flight tests and the estimated outputs which is a white noise if the system matrices are properly estimated. If the system is properly identified, the vector e is a white Gaussian noise vector with a zero mean value. The index o is the system's number of outputs. In equation (6.2.1), the vector $x(t)$ of length n represents the system's true states expressed as linear combinations between previous inputs and previous outputs.

The system's estimated states are represented by the vector $\hat{x}(t)$ and the matrix K, called the noise disturbance matrix, represents the effect of the measurement noise on the state noise. The *estimated* states $\hat{x}(t)$ are used to find the estimated outputs vectors $\hat{y}(t)$ as shown in the widely known state space system of equations (6.3.1) and (6.3.2):

$$\hat{x}(t+\Delta T)_{n\times 1} = A_{n\times n}\hat{x}(t)_{n\times 1} + B_{n\times m}u(t)_{m\times 1} \tag{6.3.1}$$

$$\hat{y}(t)_{o\times 1} = C_{o\times n}\hat{x}(t)_{n\times 1} + D_{o\times m}u(t)_{m\times 1} \tag{6.3.2}$$

where $u(t)$ represents the system's inputs assembled in a vector of size m. The A matrix is the state matrix whose rank is equal to the system order. The B matrix represents the effect on the states of each input defined in equation (6.1). The C matrix relates the outputs to the system's states. The D matrix relates the outputs to the system's inputs. The non-zero D matrix is equivalent to a system in which the inputs influence the outputs with no time delay. Since the system identified in this paper is a dynamic system, there is always a time delay between the inputs and outputs, therefore the D matrix is a *null* matrix. When the D matrix is null, the outputs $\hat{y}(t)$ are only functions of the state vectors $\hat{x}(t)$, which are written as functions of the inputs and states at the previous time step, see equation (6.3.2). If the model should be used for a flight simulation, the model error is unknown because no flight test measurement is available and only equations (6.3.1) and (6.3.2) are used to simulate the outputs. In other words, it is not necessary to take the noise covariance matrix K into account.

The A, B, C, D and K matrices terms are usually estimated by means of various parameter estimation methods. Most classical parameter estimation methods start with a set of first guesses, which can be based on physical insight of the system and iterate from these guesses to minimize the error between the model and the given data with a minimization algorithm. If the initial parameter guesses are far from their true values, the minimization algorithm may converge towards a local minimum, which is a disadvantage of this method.

In the framework of our study, we choose to use the subspace identification algorithm. Its main advantage resides in the fact that it is a non-iterative algorithm, which does not require any initial guess of the terms in the matrices [A, B, C, D, K], and therefore finds the matrices parameters solely from the known inputs and outputs. For this reason, the subspace identification method is much faster than the classical estimation methods and has no problem related to optimization, such as the possible convergence of the solution towards the local minimum instead of the global minimum. Furthermore, this method

does not require any a-priori knowledge of the system. The subspace identification algorithm is implemented with the MATLAB® System Identification Toolbox. The basic theory behind this algorithm is described in by Ljung (1999) and the manner in which the algorithm is implemented in MATLAB® is presented by Ljung (2006). The subspace system identification method has been successfully used in recent literature for different applications such as fiber optic research by Galvao (2005) and the identification of aeroelastic instabilities on an F/A-18 aircraft by Brenner (1997).

The main concept behind the subspace method is the definition of the system observability matrix Γ_r in equation (6.4) from modern control theories, where the forward prediction horizon is represented by r (Ljung, 1999). This matrix can be obtained from the system's inputs $u(t)$ and outputs $y(t)$ and its expression is as follows:

$$\Gamma_r \overset{def}{=} \begin{bmatrix} C \\ CA \\ \cdots \\ CA^{r-1} \end{bmatrix} \tag{6.4}$$

Once this observability matrix Γ_r is known, the state space matrices $[A, B, C, D, K]$ can be obtained. When the subspace method is used, the order of the state space system can be defined by the user so as to obtain better results. In general, a higher order system will give a better match for the data records used during the model identification process. However, choosing a too high order can lead to a loss of generality which may result into a degradation of the results on the records set aside for the validation process. The model orders which offer the best compromise are: 3 for the Main rotor torque, 4 for the Tail rotor torque, 2 for the Engine torque and 2 for the Main rotor speed model.

6.3.4 Refinement of the subspace identification method by use of the Levenberg-Marquardt minimization algorithm

Following the application of the subspace method, the parameter values in the state space matrices are further refined by using the Levenberg-Marquardt minimization algorithm which is widely known and is described by Jategaonkar (2006) with the parameters found by using the subspace method as the first guesses. The identification results are obtained and further compared with and without this minimization method. The cost function to be minimized is the output error which is defined by the following equation (6.5):

$$J\left(\hat{\theta}\right)=\frac{1}{2}\sum_{i=1}^{N}\left[y(t_i)-\hat{y}\left(t_i|\hat{\theta}\right)\right]^2 \tag{6.5}$$

In equation (6.5), $J\left(\hat{\theta}\right)$ is the cost function, $y(t_i)$ represents the outputs from the flight test data at time t_i, $\hat{y}\left(t_i|\hat{\theta}\right)$ represents the estimated outputs at time t_i, which depends on the model's estimated parameter values $\hat{\theta}$. In this equation, the error between the estimated outputs and outputs from flight test data is summed over the length of the record where N correspond to the number of data points in the data vector. This cost function represents the least square error between the estimated outputs (such as the main rotor torque) and the outputs from flight test for a given value of the vector $\hat{\theta}$ of estimated parameters.

In order to minimize the cost function defined in equation (6.5), it was decided to use the Levenberg-Marquardt minimization algorithm. This algorithm was chosen because it combines the advantages of two well known algorithms: the Gradient Descent algorithm and the Gauss Newton algorithm. In both the Gradient Descent and the Gauss-Newton algorithms, the $\hat{\theta}$ parameter estimate is updated at each iteration by using the following equation:

$$\hat{\theta}_{j+1} = \hat{\theta}_j + \Delta\hat{\theta}_j \tag{6.6}$$

Where the j index is the iteration number and $\Delta\hat{\theta}$ is an increment vector for each parameter $\hat{\theta}$ in the state-space matrices. The optimization algorithm determines these increments' value. For both algorithms, this increment is proportional to the cost function $J\left(\hat{\theta}\right)$ values, which means that as the algorithm converges towards a minimum, the $\Delta\hat{\theta}$ increments are reduced. Both algorithms are now explained:

6.3.4.1 Gradient Descent Algorithm

In the Gradient Descent algorithm, the gradient of the $J\left(\hat{\theta}\right)$ cost function is determined and the parameters are updated in the negative direction of this gradient, as expressed in the following equation:

$$\hat{\theta}_{j+1} = \hat{\theta}_j - \frac{\partial J\left(\hat{\theta}_j\right)}{\partial\hat{\theta}_j} \tag{6.7}$$

The second term on the right hand side of equation (6.7) is obtained by differentiating the term given by equation (6.5) with respect to $\hat{\theta}_j$ as follows:

$$\frac{\partial J\left(\hat{\theta}_j\right)}{\partial\hat{\theta}_j} = -\sum_{i=1}^{N}\left[\frac{\partial\hat{y}\left(t_i|\hat{\theta}_j\right)}{\partial\theta_j}\right]^T\left[y\left(t_i\right) - \hat{y}\left(t_i|\hat{\theta}_j\right)\right] = G \tag{6.8}$$

In equation (6.8), G represents the gradient and each parameter in the $\hat{\theta}_j$ vector at the j^{th} iteration. The parameter update can be derived from equations (6.7) and (6.8) by using the following equation:

$$\Delta\hat{\theta}_j = -\frac{\partial J\left(\hat{\theta}_j\right)}{\partial\hat{\theta}_j} = \sum_{i=1}^{N}\left[\frac{\partial\hat{y}\left(t_i|\hat{\theta}_j\right)}{\partial\hat{\theta}_j}\right]^T\left[y\left(t_i\right)-\hat{y}\left(t_i|\hat{\theta}_j\right)\right] = -G \tag{6.9}$$

This algorithm is known to be fairly robust, but may require a significant number of iterations to reach a minimum. The Gauss-Newton algorithm which usually converges faster than the Gradient Descent algorithm, and for this reason, is described below:

6.3.4.2 Gauss Newton Algorithm

The Gauss-Newton algorithm is derived from the postulate that at a minimum, the cost function gradient is zero as shown in equation (6.10):

$$\frac{\partial J\left(\hat{\theta}_j\right)}{\partial\hat{\theta}_j} = 0 \tag{6.10}$$

The value of the cost function gradient at iteration $j+1$ can be calculated by a Taylor series expansion of its value at iteration j, as shown in the following equation:

$$\left(\frac{\partial J\left(\hat{\theta}\right)}{\partial\hat{\theta}}\right)_{j+1} \approx \left(\frac{\partial J\left(\hat{\theta}\right)}{\partial\hat{\theta}}\right)_j + \left(\frac{\partial^2 J\left(\hat{\theta}\right)}{\partial\hat{\theta}^2}\right)_j \Delta\hat{\theta} \tag{6.11}$$

We set the gradient at iteration $j+1$ to zero in equation (6.11) and we isolate the term $\Delta\hat{\theta}$ yields to the following equation:

$$\Delta\hat{\theta} = -\left(\frac{\partial J^2\left(\hat{\theta}\right)}{\partial\hat{\theta}^2}\right)_J^{-1}\left(\frac{\partial J\left(\hat{\theta}\right)}{\partial\hat{\theta}}\right)_J \tag{6.12}$$

The double derivative in equation (6.12) may be evaluated from equation (6.9) in the following manner:

$$\frac{\partial J^2\left(\hat{\theta}_J\right)}{\partial\hat{\theta}_J^2} = \sum_{t=1}^{N}\left[\frac{\partial\hat{y}\left(t_t|\hat{\theta}_J\right)}{\partial\hat{\theta}_J}\right]^T\frac{\partial\hat{y}\left(t_t|\hat{\theta}_J\right)}{\partial\hat{\theta}_J} + \sum_{t=1}^{N}\left[\frac{\partial^2\hat{y}\left(t_t|\hat{\theta}_J\right)}{\partial\hat{\theta}_J^2}\right]^T\left[y\left(t_t\right) - \hat{y}\left(t_t|\hat{\theta}_J\right)\right] \tag{6.13}$$

In case when equation (6.13) was used to complete this second derivative, this method would be referred to as the Newton method. However, in the Gauss-Newton method, the second term is neglected for two reasons:

1. It requires a lot of calculations to obtain the output's second derivative
2. It tends to zero as the optimization algorithm converges since the term $\left[y\left(t_t\right) - \hat{y}\left(t_t|\hat{\theta}_J\right)\right]$ tends to zero.

Equation (6.13) therefore becomes:

$$\frac{\partial J^2\left(\hat{\theta}_J\right)}{\partial\hat{\theta}_J^2} \approx \sum_{t=1}^{N}\left[\frac{\partial\hat{y}\left(t_t|\hat{\theta}_J\right)}{\partial\hat{\theta}_J}\right]^T\frac{\partial\hat{y}\left(t_t|\hat{\theta}_J\right)}{\partial\hat{\theta}_J} = F \tag{6.14}$$

Insertion of equations (6.8) and (6.14) into equation (6.12), gives the following equation for the parameter increment in the Gauss-Newton algorithm, where F represents the second derivative:

$$F\Delta\hat{\theta} = -G \tag{6.15}$$

The Gauss-Newton algorithm is known to converge faster than the Gradient Descent algorithm, however it is very sensitive to the initial conditions and may converge towards a local minimum. A good optimization approach consists in the combination of the advantages of the Gradient Descent algorithm and the Gauss-Newton algorithm of the Levenberg-Marquardt method used in the framework of this research. The parameter vector increment in the Levenberg-Marquardt algorithm is defined by the following equation:

$$\left(F + \lambda I\right)\Delta\hat{\theta} = -G \tag{6.16}$$

From equations (6.9) and (6.15), it clearly appears that the Levenberg-Marquardt method is a linear combination of the Gradient Descent and the Gauss-Newton methods. The λ coefficient is called the Levenberg-Marquardt parameter. If its value is zero, then the algorithm is a pure Gauss-Newton algorithm; if its value is infinite, then the algorithm is a pure Gradient Descent algorithm. This parameter varies as the optimization is carried on so as to ensure the fastest possible convergence rate.

The subspace method was used to obtain the initial guesses of the state-space models parameters. The Levenberg-Marquardt method was further used to refine these parameters' values. The results will show the model error with and without this optimization. The implementation method presented in figure 6.2 is valid for the model's implementation in a flight simulator. As discussed previously, these limits parameter models could also be used in a *carefree* envelope protection control system. In this case, the model implementation would be different than the one presented in figure 6.2 - and will be discussed in the following section.

6.3.5 Implementation of the prediction model

In the case of a carefree envelope protection control system, it is necessary to limit the future value of limit parameters. Let us recall that in this paper, the limit parameters are the main rotor torque, tail rotor torque, engine torque or main rotor speed. During the helicopter's flight, the limit parameter's current value, as well as the helicopter states and control inputs are known from flight tests (measurements). In order to prevent any limit violation, it is necessary to predict the future values of the helicopter's limit parameters for a prediction horizon. According to Jeram (2002), the required prediction horizons for a cueing application lie between 0.25 and 0.5 seconds. When the prediction horizon increases, the model error is also increased, however, a cueing system has more time to warn the pilot of the incoming limits. The appropriate prediction horizon should ultimately be determined by the pilot during a flight test program. The model used to predict the future values of limit parameters is illustrated in figure 6.3:

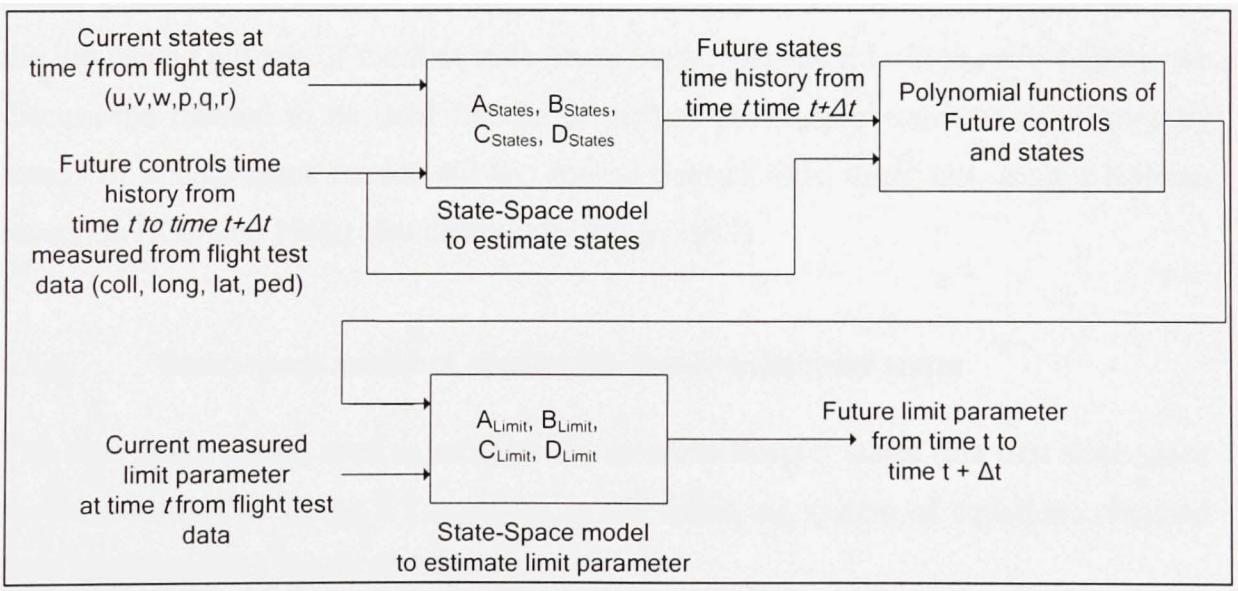

Figure 6.3 ***Model structure used for the prediction of the future value of a limit parameter (main rotor torque, tail rotor torque, engine torque or main rotor speed).***

As shown in figure 6.3, two state space models are necessary to predict the future value of the helicopter's limit parameters. The first state space model uses the current states from flight test data and an assumed time history of the control inputs to predict the future time histories of the helicopter states [u, v, w, p, q, r] within a given prediction horizon. In practice, according to Jeram (2002) the assumed future value of the pilot control inputs can be either a worse case scenario or more commonly, similar to the control inputs used to train the model. In this paper, in order to accurately measure the model's performance, it was decided to assume that the future controls were the same as in the flight test data. Further research will be conducted in the future so as to identify a reasonable procedure to estimate the future control time history. The future states' time history is combined with the future value of the pilot's control inputs in a polynomial equation to construct the inputs of the limit parameter models (such as the main rotor torque models) as in equation (6.1). The second state-space model in figure 6.3 is used to predict the future time history of a given limit parameter. This second state-space model is actually the same as the one presented in figure 6.2. The next sections deal with the inputs and outputs of the first state-space model presented in figure 6.3. Finally, we discuss the method to be used for the prediction of a future outputs time history by means of a state-space model and the current outputs from flight test using a Kalman recursion (Kalman, 1960) also detailed by Rugh (1993).

6.3.6 State-space model to predict the future helicopter states

The state space model used to estimate the future helicopter states (the first state-space model presented in figure 6.3 is shown in the following system of equations obtained from equations (6.3.1) and (6.3.2):

$$\begin{bmatrix} \hat{u}(t+\Delta t) \\ \hat{v}(t+\Delta t) \\ \hat{w}(t+\Delta t) \\ \hat{p}(t+\Delta t) \\ \hat{q}(t+\Delta t) \\ \hat{r}(t+\Delta t) \end{bmatrix} = A_{States} \begin{bmatrix} \hat{u}(t) \\ \hat{v}(t) \\ \hat{w}(t) \\ \hat{p}(t) \\ \hat{q}(t) \\ \hat{r}(t) \end{bmatrix} + B_{States} \begin{bmatrix} coll(t) \\ long(t) \\ lat(t) \\ ped(t) \end{bmatrix} \tag{6.17.1}$$

$$\begin{bmatrix} \hat{u}(t) \\ \hat{v}(t) \\ \hat{w}(t) \\ \hat{p}(t) \\ \hat{q}(t) \\ \hat{r}(t) \end{bmatrix} = C_{States} \begin{bmatrix} \hat{u}(t) \\ \hat{v}(t) \\ \hat{w}(t) \\ \hat{p}(t) \\ \hat{q}(t) \\ \hat{r}(t) \end{bmatrix} + D_{States} \begin{bmatrix} coll(t) \\ long(t) \\ lat(t) \\ ped(t) \end{bmatrix} \tag{6.17.2}$$

In equations (6.17.1) and (6.17.2), the states which are the helicopter's linear and angular velocities in body axes, are estimated with an approximate linear state-space system. The term Δt represents the simulation's sample time. Since the outputs and the states are the same, the C_{States} matrix is the identity matrix, while the D_{States} matrix is a null matrix. Even though this linear model is very approximate to estimate the future helicopter states, it has been found to be accurate enough in the overall model to provide a proper prediction of the limit parameter's future values within a given prediction horizon.

6.3.7 Future outputs prediction based on the current outputs from flight test data and a state-space model

In this section, we will implement a state space model which jointly uses the outputs from flight test data and the future inputs time history to predict the future outputs time history. This procedure was first developed by Kalman [21] and is widely used in

control theory [20]. We first demonstrate the one time step prediction, and then the multiple time step prediction.

6.3.7.1 One time step ahead prediction

In this case, the output's current value is obtained from the flight test data. By knowing the flight test outputs $y(t)$ and the estimated outputs $\hat{y}(t)$, is possible to determine the measurement error with the following equation derived from equation (6.2.2), where the D matrix can be neglected since is a null matrix:

$$v(t) = y(t) - \hat{y}(t) = y(t) - C\hat{x}(t) \tag{6.18}$$

This error should correspond to an uncorrelated perturbation with the inputs. From equations (6.18), (6.3.1) and (6.2.1), is possible to obtain a new state space equation as follows:

$$\hat{x}(t+\Delta t) = A\hat{x}(t) + Bu(t) + K\left[y(t) - C\hat{x}(t)\right] \tag{6.19}$$

Equation (6.19) is known as the Kalman state observation equation (Kalman, 1960). By rearranging equation (6.19) and adding equation (6.3.2), while neglecting the D matrix, we obtain the following equation:

$$\begin{aligned} \hat{x}(t+\Delta t) &= \left[A - KC\right]\hat{x}(t) + \left[K \quad B\right]\left[y(t) \quad u(t)\right] \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{6.20}$$

where $y(t)$ represents the outputs from flight test and $\hat{y}(t)$ represents the estimated outputs. Equation (6.20) applies when the outputs need to be predicted one time step in

advance or, in other words, with a prediction horizon of one time step. Note that the length of one time step is equal to the flight test data sampling rate.

6.3.7.2 Prediction with a higher horizon than one time step

It is possible to predict the outputs further in time by using a larger prediction horizon. In this case, a pure simulation model such as the one described in equation (6.3) is run using the last prediction found by equation (6.20) and the time length of the simulation's correspond to the prediction horizon. This scheme is best illustrated by an example or by a 2 time step prediction. If one wants to predict the value of the outputs at time 3, knowing only the flight test states x and inputs u at time 2 when time step is equal to 1, is recommended to use equation (6.3) as follows:

$$\hat{y}(3) = Cx(3) \tag{6.21.1}$$

where

$$x(3) = Ax(2) + Bu(2) \tag{6.21.2}$$

In these equations, the numbers in parentheses correspond to the time samples. Equations (6.21.1) and (6.21.2) may be combined to give:

$$\hat{y}(3) = C\left[Ax(2) + Bu(2)\right] = CAx(2) + CBu(2) \tag{6.22}$$

The state $x(2)$ is unknown, but can be estimated by use of equation (6.20) where $t + \Delta t = 2$, as follows:

$$x(2) = \left[A - KC\right]x(1) + \left[K \quad B\right]\left[y(1) \quad u(1)\right] \tag{6.23}$$

where $y(1)$ are the outputs from flight test data 2 steps ahead the prediction, $u(1)$ is the system input's initial value and $x(1)$ is the initial state. The same reasoning may be extended to any prediction horizon. The general equation used to predict the value of the

outputs $\hat{y}(t)$ using the outputs measured from flight test $y(t-r)$ (prediction horizon of r) is:

$$\hat{y}(t)=\begin{bmatrix} CB & CAB & \dots & CA^{r-2}B \end{bmatrix}\begin{bmatrix} u(t-r+1) \\ \dots \\ u(t) \end{bmatrix}+CA^{r-1}x(t-r+1) \tag{6.24}$$

Where the last term $x(t-r+1)$ is found with the following equation:

$$x(t-r+1)=[A-KC]x(t-r)+[K \quad B]\begin{bmatrix} y(t-r) & u(t-r) \end{bmatrix} \tag{6.25}$$

These equations are simply the extensions of Equations (6.21) and (6.22) for a r prediction horizon. The limit case of a prediction is found when the horizon tends to infinity. In this case, the prediction outputs are the same as the simulation outputs.

6.4 Results

The results obtained for the model simulation and prediction are now presented in following two sub-sections. In the first sub-section, we present the time histories of the outputs measured from light test and estimated outputs for a typical simulation case. In the second sub-section, quantitative results are presented and the performances of the simulation and the predictions for different prediction horizons are compared. In this section we compare the results obtained by use of the Subspace system identification method only to those obtained by use of the same identification method improved with the Levenberg-Marquardt optimization algorithm.

6.4.1 Typical simulation model outputs for different manoeuvres

The results presented in figures 6.4 – 6.7 were obtained from the model simulation (see figure 6.2) where the u,v,w,p,q,r rotorcraft states are available from flight tests and the

estimated outputs are simulated at each time step. In other words, in the following figures, the prediction horizon is infinite. In each figure, the lines in the middle correspond to the model outputs and the lines around it correspond to the tolerance band around the value from flight test data.

The tolerance band is set to 3% of the *maximum* torque and was selected according to the guidelines of the FAA flight simulator qualification advisory circular (1994). Since the FAA does not specify the tolerance band neither for the Engine Torque nor the Tail Rotor Torque, a tolerance band of 3% is used in this paper for these quantities. For the Main Rotor speed, the FAA usually specifies a tolerance band of 1.5%, but since the error between the model outputs and the data outputs is minor, the tolerance band for the Main Rotor speed was set to 0.5% in the following figures.

Figures 6.4 – 6.7 present the results for collective input excitations, while figures 6.8 - 6.11 present the results for (ii) Longitudinal cyclic excitation. figures 6.12 – 6.15 present results for lateral cyclic excitation and figures 6.16 – 6.19 show results for pedal excitation. For each excitation, the time histories of the main rotor torque, tail rotor torque, engine torque and main rotor speed are displayed.

The results shown in the following figures are obtained from a model identified with the Subspace System identification method and further optimized with the Levenberg-Marquardt minimization algorithm. These results are obtained for the second flight condition characterized by 49 records used to generate the model and 20 records used to validate it (as seen in Table 6.1). In this section, we show results obtained for a number of 4 records out of 20 records used for validation.

Figures 6.4 – 6.7 show the outputs time histories following a 2311 multi-step input collective control for the next flight condition:

Mean Altitude = 9323 ft

Gross Weight GW = Heavy

Center of Gravity CG position = Aft

Mean True Airspeed TAS = 113 knots

Rate of Climb ROC = -728 ft/min.

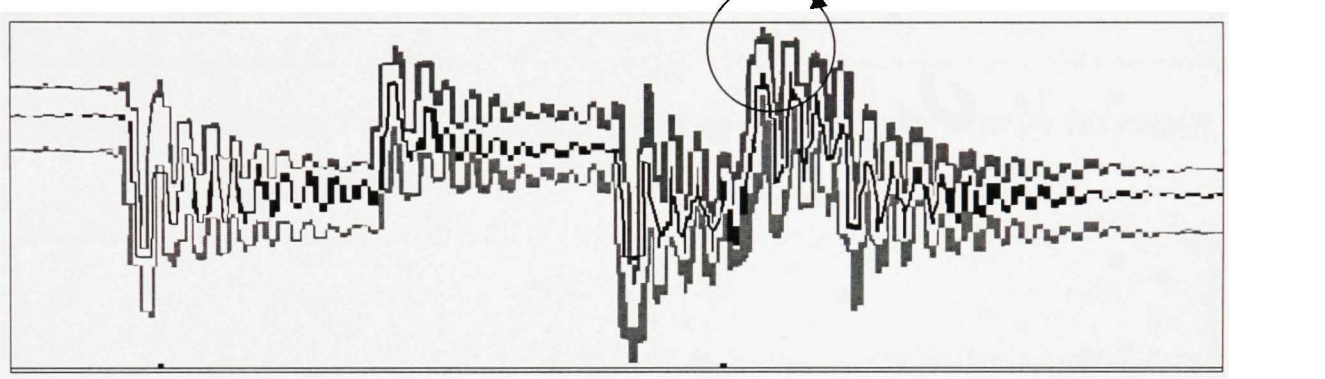

Figure 6.4 ***Main rotor torque time history following a collective 2311 input (3% tolerance band)***

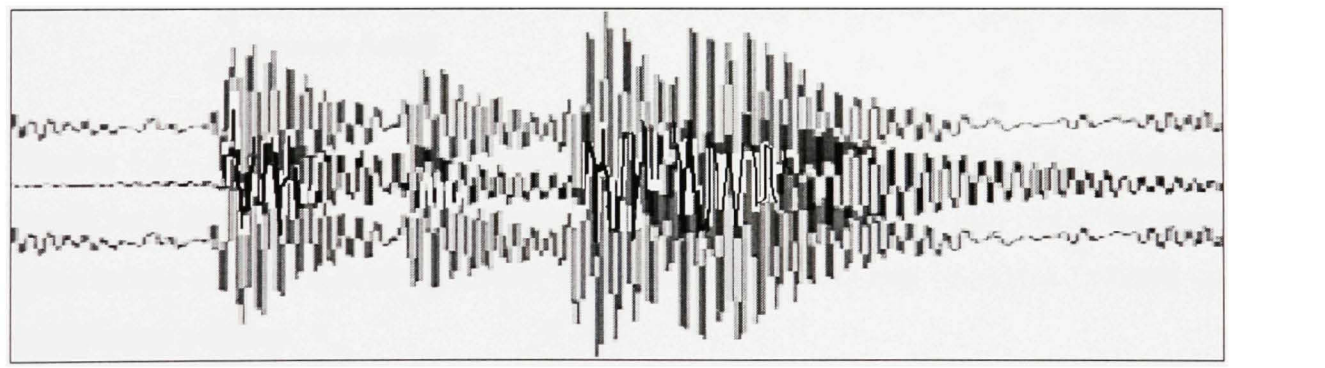

Figure 6.5 ***Tail rotor torque time history following a collective 2311 input (3% tolerance band)***

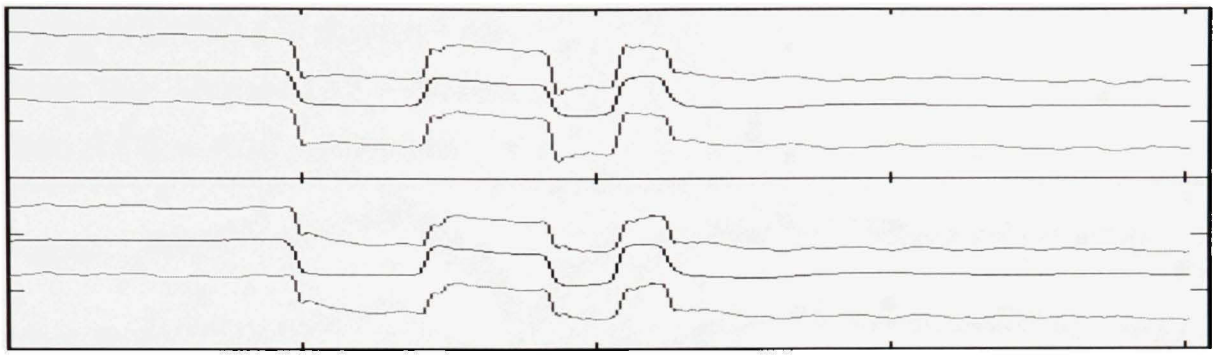

Figure 6.6 ***Engines 1 and 2 torques time histories following a collective 2311 input (3% tolerance band)***

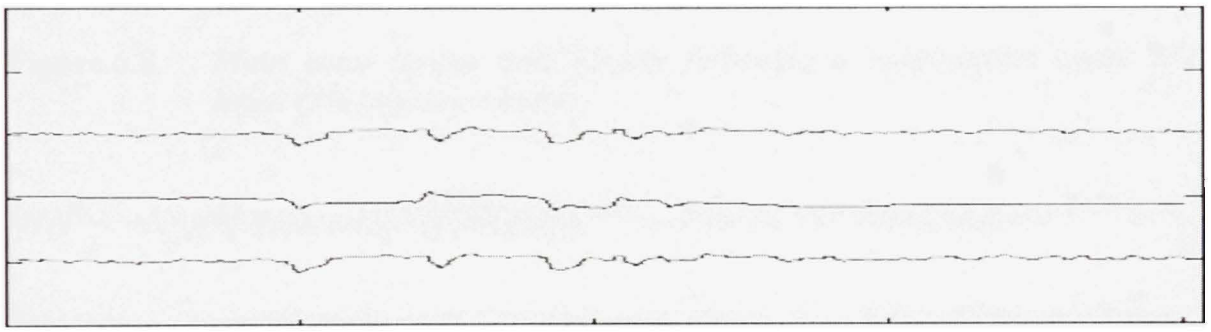

Figure 6.7 ***Main rotor speed time history following a collective 2311 input (0.5% tolerance band)***

Figures 6.4 – 6.7 clearly show that the simulated outputs lay within the FAA tolerance bands for a simulation. One can also see that following a collective step input, the main rotor torque reaches a peak at certain value (see figure 6.4) and oscillates towards an equilibrium position.

In some cases, a larger error may appear for a few hundredths of a second if the model outputs are slightly out of phase with respect to the flight tests data. Figures 6.8 – 6.11 display the same types of outputs following a longitudinal cyclic 2311 multi-step input for the following flight tests condition:

Mean Altitude = 9167 ft

Gross Weight GW = Heavy

Center of Gravity CG position = Aft

Mean True Airspeed TAS = 79 knots

Rate of Climb ROC = 367 ft/min.

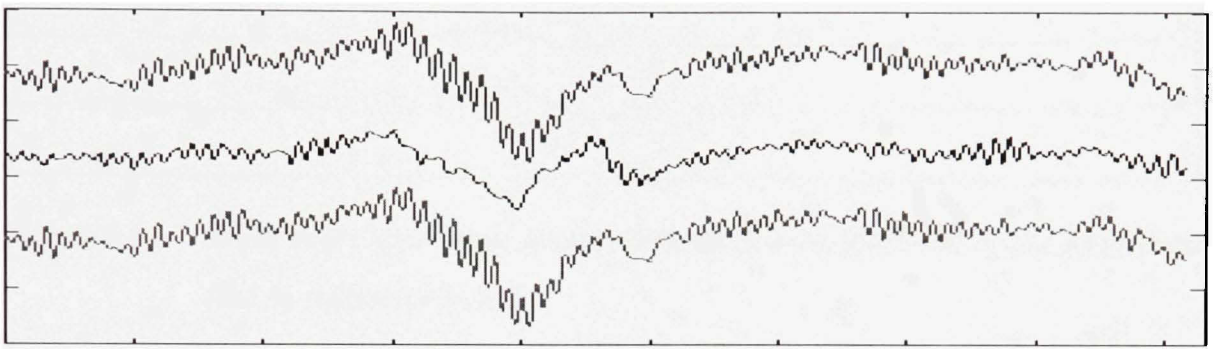

Figure 6.8 ***Main rotor torque time history following a longitudinal cyclic 2311 input (3% tolerance band)***

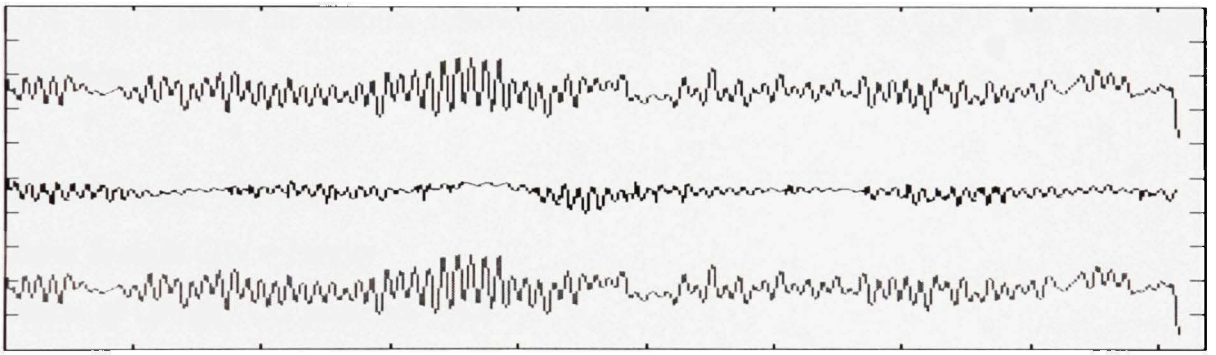

Figure 6.9 ***Tail rotor torque time history following a longitudinal cyclic 2311 input (3% tolerance band)***

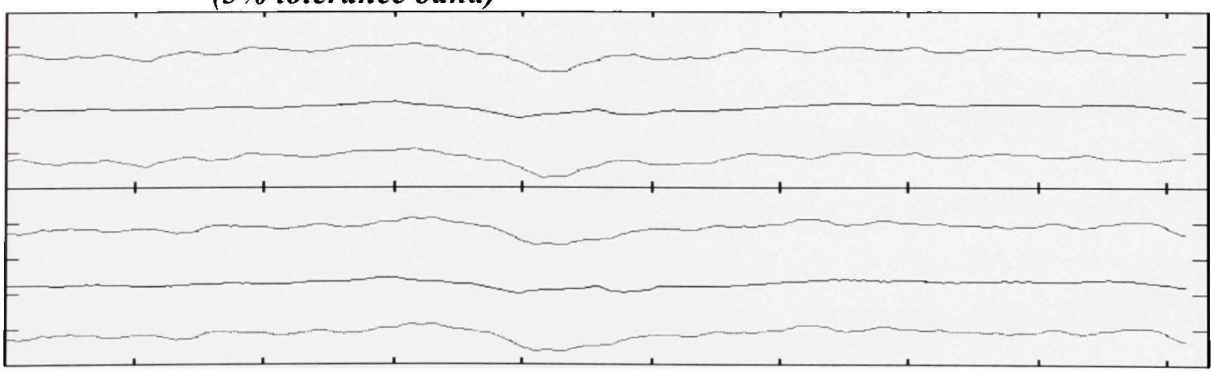

Figure 6.10 ***Engine torque time history following a longitudinal cyclic 2311 input (3% tolerance band)***

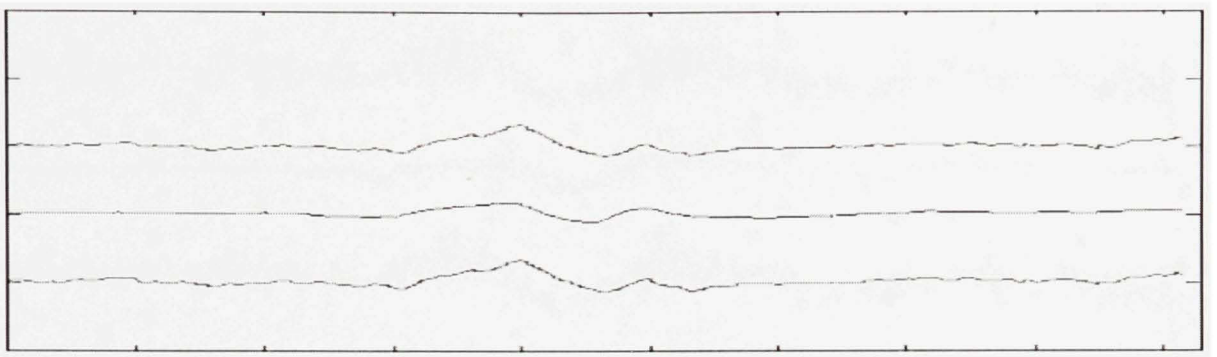

Figure 6.11 ***Main rotor speed time history following a longitudinal cyclic 2311 input (0.5 % tolerance band)***

Figures 6.8 – 6.11 clearly show that the model is within the tolerance bands. Figures 6.12 – 6.15 show the outputs following a lateral cyclic 2311 input for the next flight condition:

Mean Altitude = 9581 ft
Gross Weight GW = Heavy
Center of Gravity CG position = Aft
Mean True Airspeed TAS = 71 knots
Rate of Climb ROC = 915 ft/min

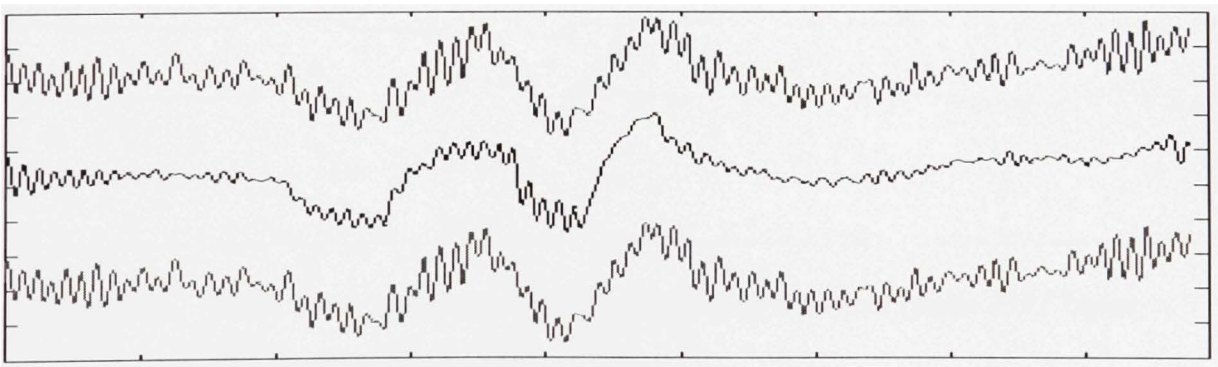

Figure 6.12 ***Main rotor torque time history following a lateral cyclic 2311 input (3% tolerance band)***

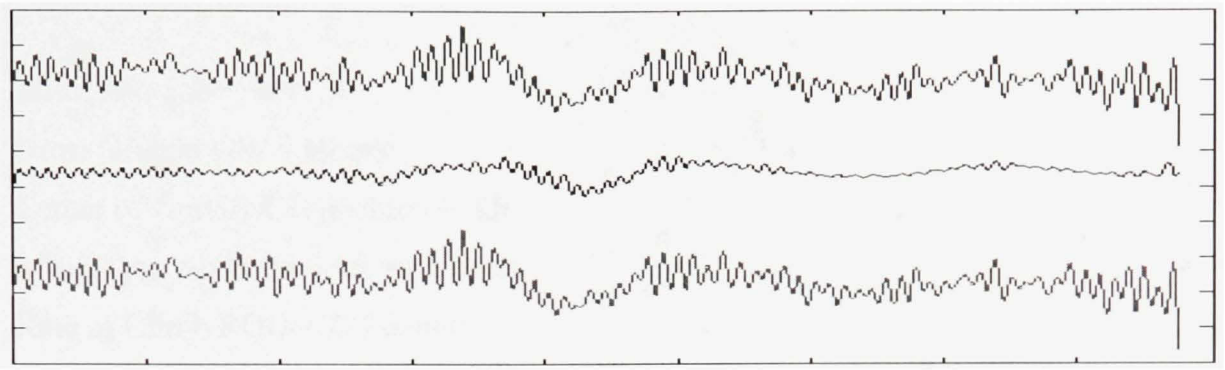

Figure 6.13 *Tail rotor torque time history following a lateral cyclic 2311 input (3% tolerance band)*

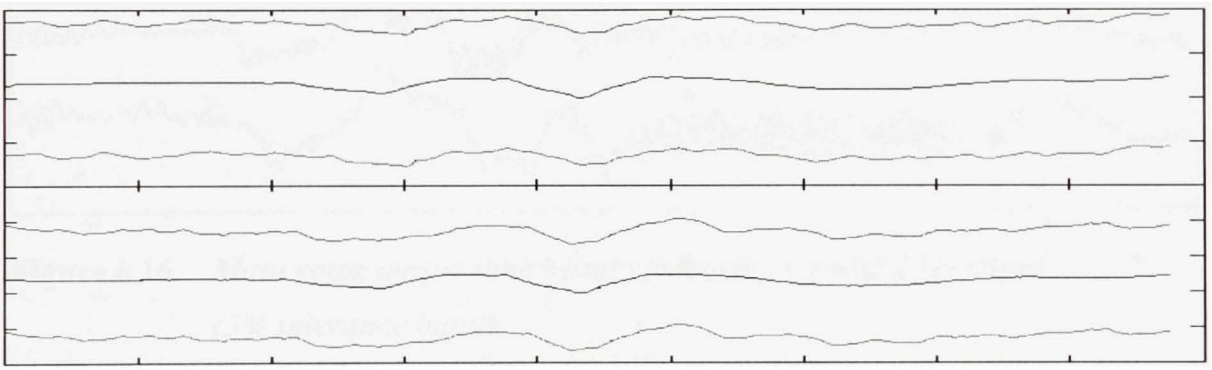

Figure 6.14 *Engine torque time history following a lateral cyclic 2311 input (3% tolerance band)*

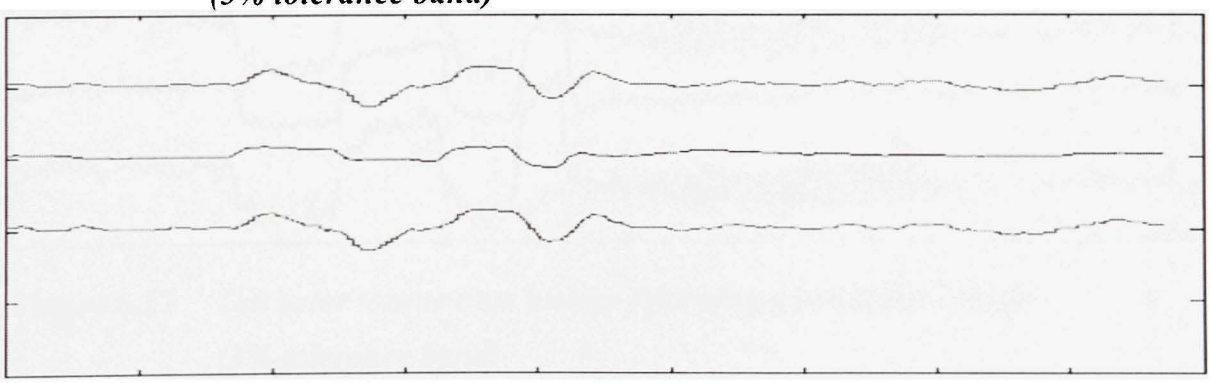

Figure 6.15 *Main rotor speed time history following a lateral cyclic 2311 input (0.5% tolerance band)*

Finally, the outputs following a pedal 2311 input are shown in figures 6.16 – 6.19 for the following flight condition:

Mean Altitude = 9011 ft

Gross Weight GW = Heavy

Center of Gravity CG position = Aft

Mean True Airspeed TAS = 81 knots

Rate of Climb ROC = 277 ft/min

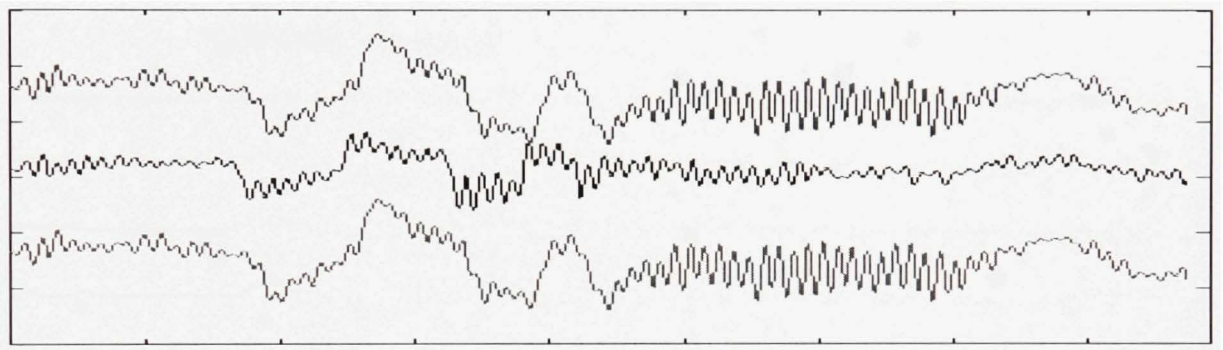

Figure 6.16 ***Main rotor torque time history following a pedal 2311 input (3% tolerance band)***

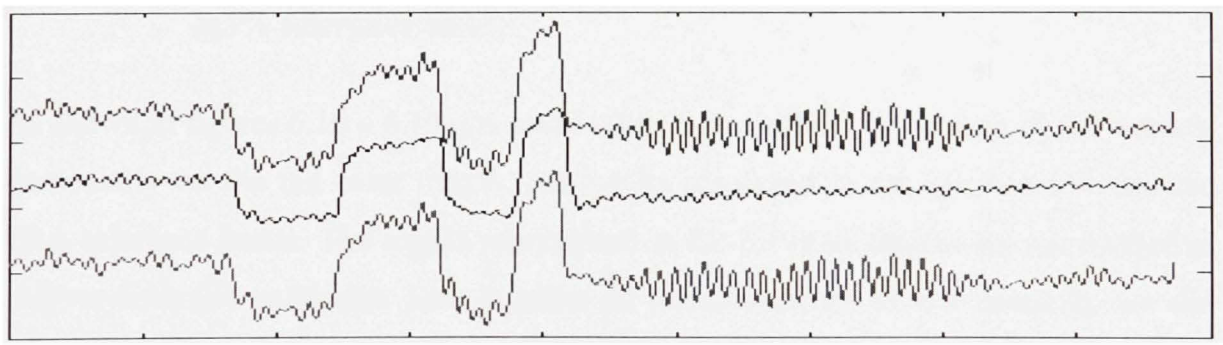

Figure 6.17 ***Tail rotor torque time history following a pedal 2311 input (3% tolerance band)***

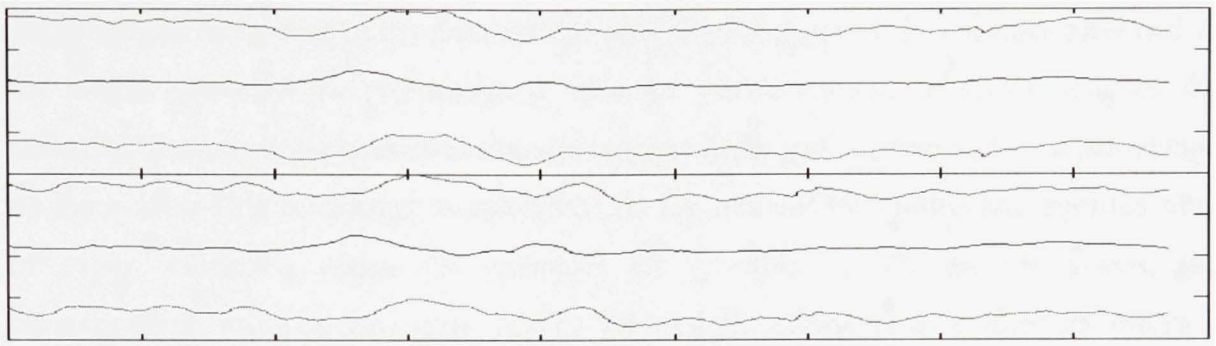

Figure 6.18 ***Engine torque time history following a pedal 2311 input (3% tolerance band)***

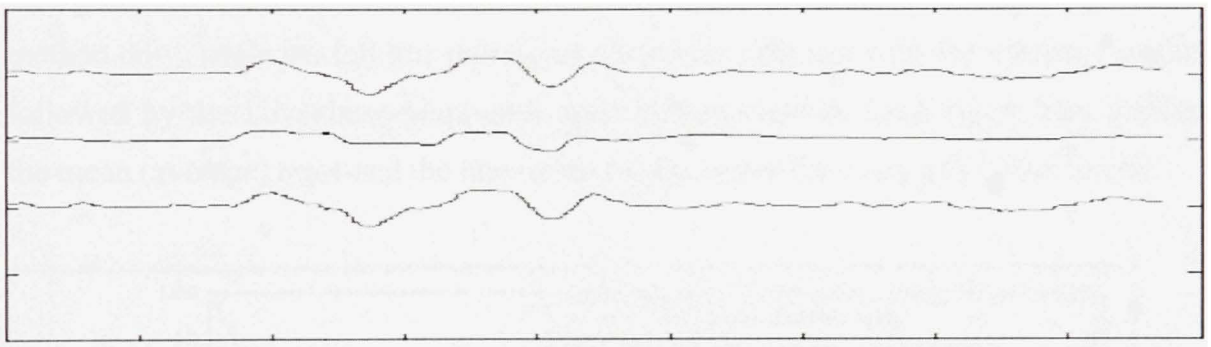

Figure 6.19 **Main rotor speed time history following a pedal 2311 input (3% tolerance band)**

As shown in figures 6.16 – 6.19, the pedal inputs have influences on each of the outputs, particularly on the tail rotor torque. All results displayed so far are found within the FAA tolerance bands. The results represented in the previous figures are representative only to four (4) validation time records. In the framework of our research, we did however validate the model by using 138 validation records. The quantitative results of these records, as well as the effects of using different prediction horizons are discussed in details in the following section.

6.4.2 Quantitative simulation model performance

This section summarizes the results obtained for the 138 validation records time histories. Figures 6.20 – 6.23 show the results for a pure simulation that was

implemented according to the architecture presented in figure 6.2. The data provided in the graphs represent the percentage of time for which the sum of errors between the simulated outputs and the measured outputs from flight test is *greater* than a percentage on the *x*-axis. This percentage is calculated on the basis of the cumulated recorded time of every validation cases, for example, for a value of 3% on the x-axis, the corresponding value on the y-axis will be the total percentage of time when the errors is higher than 3% over the total amount of time represented in every records.

The dashed line represents the results obtained with the subspace system identification method only, while the full line represents the results obtained with the subspace method followed by the Levenberg-Marquardt optimization method. Each figure also displays the mean (average) error and the maximum (worse) error for every validation record.

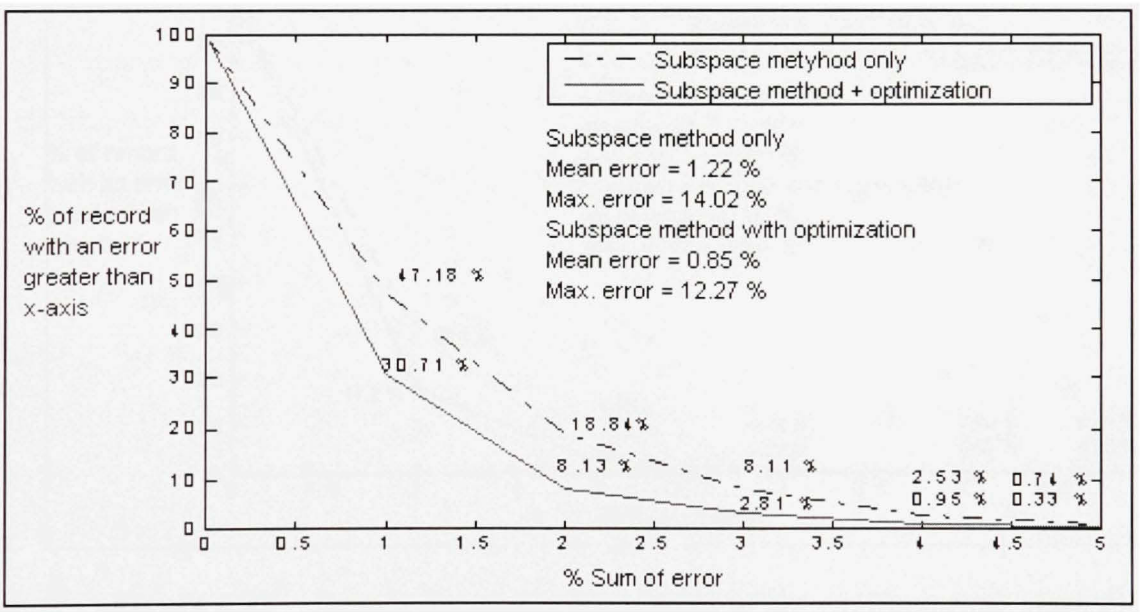

Figure 6.20 ***Results for a pure simulation of the main rotor torque output***

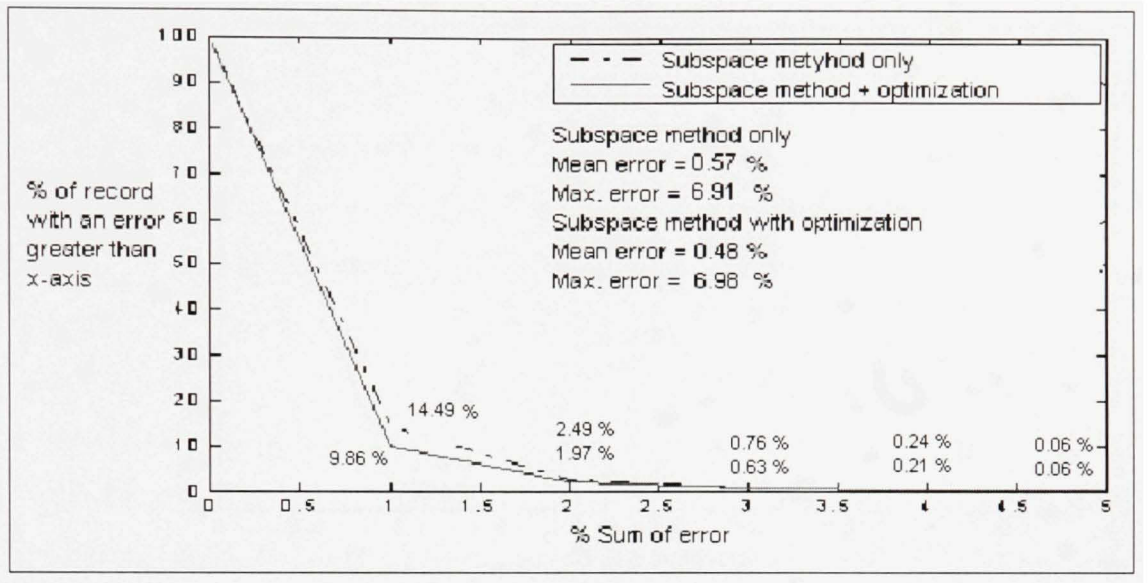

Figure 6.21 ***Results for a pure simulation of the tail rotor torque output***

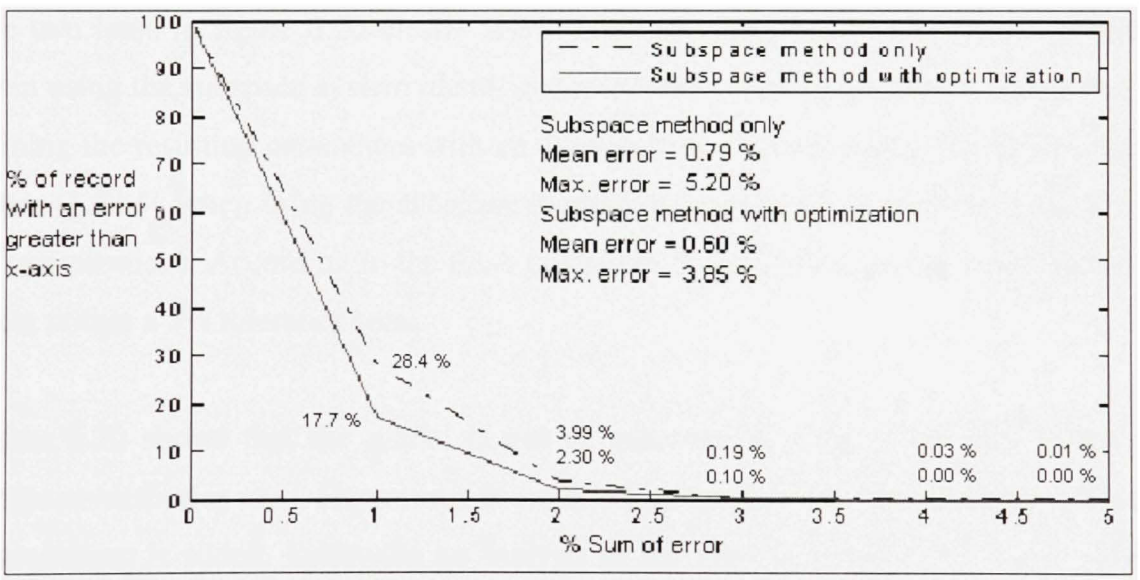

Figure 6.22 ***Results for a pure simulation of the engine torque output***

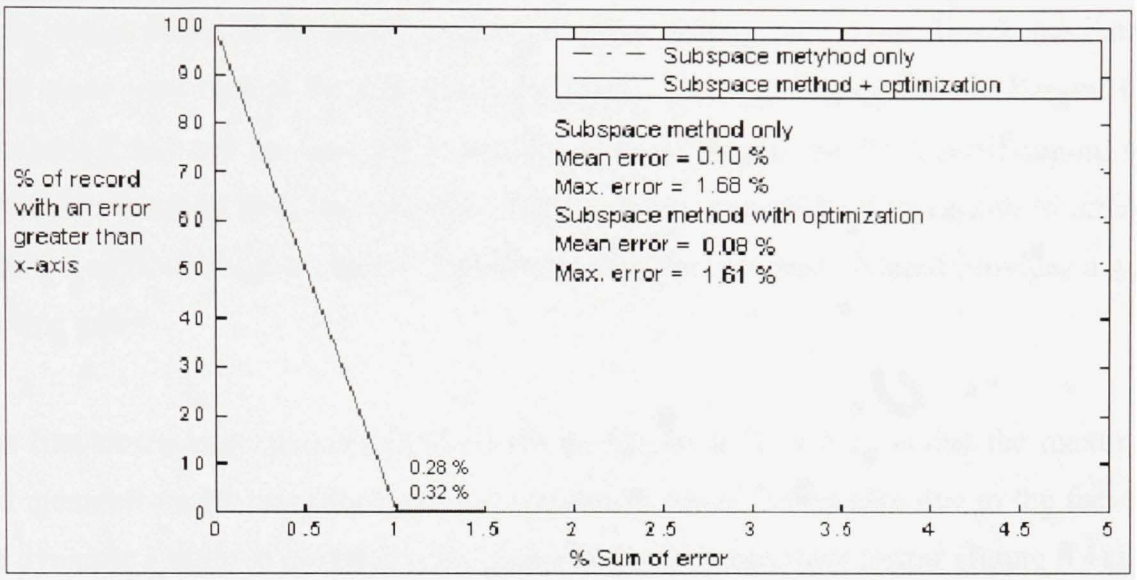

Figure 6.23 ***Results for a pure simulation of the main rotor speed output***

The two lines in figure 6.20 clearly show that even though the results are reasonable when using the subspace system identification method, they can further be improved by refining the resulting parameters with an optimization. In both cases, the mean error is minor (1.22 % when using the subspace method only and 0.85 % when refining it with an optimization). According to the FAA guidelines, a simulation model is satisfactory if it lies within a 3% tolerance band.

Figure 6.20 shows that the model is out of tolerance 8.11 % of the time when the subspace method is used. However it is out of tolerance only 2.8 % of the time when the optimization is added. Following an analysis of the lines in figure 6.20, it also clearly appears that most of the time, the model does not exceed the FAA's tolerance bands; the error exceeds 5 % only 0.33 % of the time when the subspace method is used together with an optimization, which means that even though the maximum error seems to be high, it only occurs for a very short period of time.

According to our observations, the maximum error usually occurs during the collective input time history (see figure 6.20)) that is, when high frequency oscillations in the main

rotor torque occur. If the model output becomes slightly out of phase with the data, a large error may appear for a few hundredths of a second during the oscillation. One conclusion that can be drawn from these results is that for the FAA certification, it is preferable to apply both the subspace method and an optimization algorithm in order to obtain a sufficiently good model. However, using the subspace method provides a good starting point.

The first observation that can be drawn from figures 6.21 to 6.23 is that the maximum and mean errors decrease (or improve) constantly which is probably due to the fact that the dynamic system to model is more complex for the main rotor torque (figure 6.4) than for the other outputs because it has high frequencies oscillations and a damping varying with time.

The tail rotor torque and engine torque outputs are shown in figures 6.21 – 6.23, where a tolerance band of 3% is assumed, and we found that there is no difference between out of tolerance records percentage when using the subspace method only and out of tolerance record percentage when also resorting to an optimization. The same remark is true for the main rotor speed (figure 6.23)) where, in this case, the FAA tolerance band is 1.5%, which means that the subspace method is probably sufficient to model these outputs. It can also be observed from figures 6.20 to 6.23 that the full line and the dashed line are separated at the left of the plots and converge to the right of the plots. This means that if the tolerance band was set very tight (for example, 1%), there would be more significant benefit in term of the amount of time the model spends out of tolerance in improving the estimates of the parameters with an optimization algorithm than if the tolerance band was higher. Therefore, combination of an optimisation to the subspace method allows the improvement (minimization) of the error mainly when is already low and the optimisation has a lesser influence upon the parts of the records with a larger error. In the following section, we will analyse the performance of the above-discussed system when a prediction method is used.

6.4.3 Quantitative prediction model performances

Figures 6.24 – 6.27 (each figure has two parts: upper and lower), are drawn for the same flight conditions as the ones for figures 6.20 – 6.23, and show the model's performance when implemented as a prediction tool. For each helicopter output, the mean error (upper part) and the overall percentage (lower part) of the records time histories for which the error is greater than 3% are plotted with respect to the prediction horizon. The model's performance was evaluated for prediction horizons of 0.12, 0.26 and 0.5 seconds, which are all multiples of the data sampling rate. The results of a pure simulation are also presented in each figure so as to compare them with the prediction results.

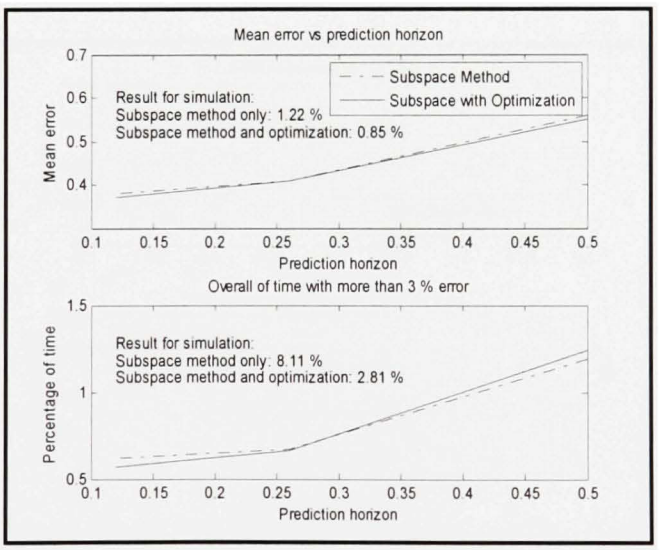

Figure 6.24 ***Mean error prediction for the main rotor torque and percentage of records with an error rate greater than 3 % with respect to the prediction horizon***

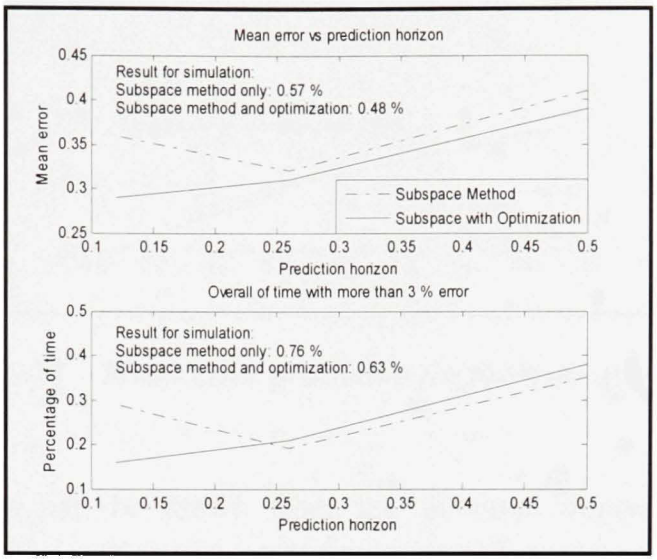

Figure 6.25 ***Mean error prediction for the tail rotor torque and percentage of records with an error rate greater than 3% with respect to the prediction horizon***

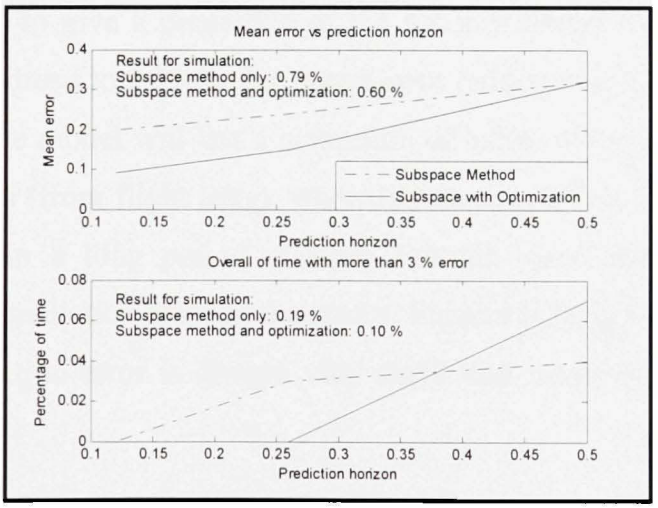

Figure 6.26 ***Mean error prediction for the engine torque and percentage of records with an error rate greater than 3% with respect to the prediction horizon***

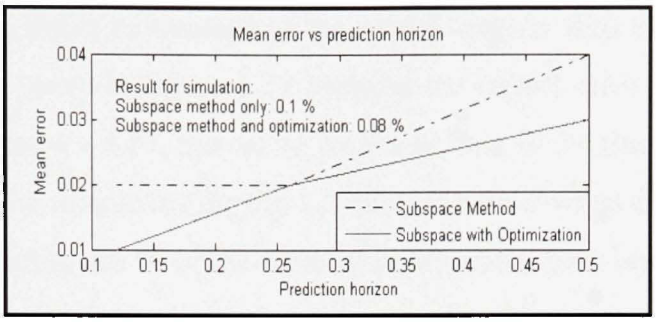

Figure 6.27 ***Mean error prediction for the main rotor speed***

Several observations can be drawn from the previous figures. In each case, as anticipated, the mean error decreases when the prediction horizon is decreased. Since a simulation is mathematically equivalent to a prediction with an infinite horizon, all the prediction results are better than the simulation results. For example, the main rotor torque model identified with the subspace method and further refined with an optimization has a mean simulation error of 2.81 % (figure 6.24), but when the state-space model is used to give a prediction of 0.5 seconds ahead from known flight test data, the mean error drops to 0.39 %. The mean error reduction is due to the fact that any prediction error in the model will last a maximum of half a second before its correction by the measurements (from flight tests), whereas in a simulation, a modelling error can affect the outputs for a long period of time. For the same reason, the mean error decreases when the prediction horizon decreases. Figures 6.24 to 6.27 also clearly show that the mean prediction error is always very small and never exceeds 0.5 % for any prediction horizon.

The second curve in each figure (except for figure 6.27 in which the error is zero) represents the percentage of time for which the model error is greater than 3 %. This percentage was chosen because it was used in the discussion above on simulation application of the model. For control system applications, there is however no fixed tolerance. The evaluation is rather based on the overall control system's performance, which is assessed by the pilot. Figures 6.24 to 6.26 reveal that a decreasing prediction

horizon results into a lower percentage of the model outputs with an error greater than 3 %. There is no such curve in figure 6.27 because the model error is always below this tolerance. In figures 6.24 – 6.27, the worse results pertain to the main rotor torque whose prediction is out of the tolerances during 1.2 % of the time when the prediction horizon is 0.5 seconds and during 0.6 % of the time when the prediction horizon is 0.12 seconds (figure 6.24).

Finally, each figure shows that both the models identified with the subspace method only and with the subspace method followed by an optimization, give excellent results. In the case of the main rotor torque outputs, there is almost no difference between the models identified with these two methods. Recall that for a pure simulation (see figure 6.20)), a significant model performance gain is realized by use of an optimization. For the tail rotor torque outputs (see figure 6.25)), resorting to an optimization allows to slightly reduce the model's mean error, but oddly, it is out of tolerance a little longer than with the sole use of the subspace method, when the prediction horizon is high. The error also slightly increases when a 0.12 seconds prediction horizon is used, with respect to the error calculated with a 0.26 seconds prediction horizon (figure 6.27).

The use of an optimisation slightly improves the main error between the model and the data for the engine torque and main rotor speed outputs however it may affect the percentage of records with an error greater than 3 %.

6.5 Discussion

The results allow us to draw the conclusions:

6.5.1 For the model simulation implementation (see figure 6.2)

1. All the simulation results, especially the main rotor torque outputs, were improved when the subspace identification method was combined with an optimization. For the main rotor torque outputs which are the worse result, 2.81% (smaller than 3%) of the simulated outputs were out of tolerance instead of 8.11% when the subspace method alone was used.

2. The addition of an optimization to the subspace method may reduce the mean error, however the benefits of an optimization decrease when the subspace method error is already small, which is especially true for the main rotor speed.

6.5.2 For the model prediction implementation (see figure 6.3)

1. The mean error and the percentage of time when the error is greater than 3% are very low for a prediction. Furthermore, as could be expected, the prediction error decreases when the prediction horizon decreases.

2. The parameters' values are refined by the subspace method with an optimization, and therefore allow the improvement of the results. The errors are already found to be very low by use only of the subspace method, which is the reason for which optimization is not needed.

6.6 Conclusions

In summary, it can be concluded that the dynamics of the main rotor torque, tail rotor torque, engine torque and main rotor speed can be properly estimated by using a state-space model. The subspace system identification method is an efficient non iterative method that can provide a reasonably good estimation of these helicopter parameters without requiring any initial guess or any prior knowledge of the system's dynamics. The model identified with the subspace method can also be successfully refined with a Levenberg-Marquardt minimization algorithm.

The results were assessed according to the mean error and the percentage of times that the model error was greater than a certain tolerance. For a flight simulation certification, the FAA's tolerance band is 3% for the main rotor torque and 1.5 % for the main rotor speed. Since there currently is no specified tail rotor torque tolerance, nor any specified engine torque tolerance, we decided to use a 3 % tolerance band. The results show that for the tail rotor torque, the engine torque and the main rotor speed, the modelling error is very small when the subspace identification method is used alone. The results for the main rotor torque are reasonable, except for 8.11 % of the records which have an error greater than 3%. This percentage was successfully reduced to 2.81 % with an optimization. Notice that the simulation results shown in this paper are for the records set aside for the validation, the results for records used for the identification are generally slightly better with mean errors up to 25 % lower. This is not surprising since these records were used to generate the model.

The model was also tested for a prediction implementation. In each case, the modelling error was very low and was reduced as the prediction horizon length was shortened. Using an optimization allowed to slightly improve the results achieved with the subspace method, but the benefit of adding an optimization was minor since the models

identified with the subspace method were already excellent. The prediction results that are discussed in this paper also applies apply for the records set aside for the validation. The mean errors for the records used in the identification are usually about the same or slightly better (up to 5 %).

In conclusion, the subspace method is a promising method to estimate helicopter's physical parameters, by means of flight test data, for simulation and control applications.

6.7 Acknowledgements

This work was possible due to the research funds received from *Bell Helicopter Textron* and the *Consortium for Research and Innovation in Aerospace in Quebec* (CRIAQ) in the CRIAQ 3.4 project entitled *Development of a new parameter estimation technology for a global helicopter model.* Thanks are addressed to main members of our team, mainly to Mr Joey Seto and Mr Ed Lambert from Bell Helicopter Textron and to Mr Ken Hui from the *National Research Center* NRC.

6.8 References

Brenner Martin, Eric Feron, April 1997, "Wavelet analysis of F/A-18 aeroelastic and aeroservoelastic flight test data", 38th AIAA/ASME/ASCE/AHS/ASC Structure, Structural Dynamics and Material Conference Exhibit, Kissimmee, Florida, 7-10 April, 1997., pp. 691-713

Federal Aviation Administration, 1994, *Helicopter Simulator Qualification*, Advisory Circular AC 120-63, Federal Aviation Administration, US department of transportation

Galvao, Roberto K. H., Sillas Hadjiloucas, Victor M. Becerra and John W Bowen, "Subspace system identification framework for the analysis of multimoded propagation of THz-transient signals", Institute of Physics Publishing, Measurement Science and Technology, Vol. 16, pp. 1037-1053, 2005.

Horn, Joseph, May 1999, *Flight Envelope Limit Detection and Avoidance*, PhD Thesis, Atlanta Georgia Institute of Technology, 110 p.

Horn, Joseph, Calise, A.J., Prasad, J.V.R., October 2002, "*Flight Envelope Limit Detection and Avoidance for Rotorcraft*", Journal of American Helicopter Society, Volume 47, Issue 4, pp. 253-262

Horn, Joseph, Nilesh Sahani, 2004 , "Detection and Avoidance of Main Rotor Hub Moment Limits on Rotorcraft", Journal of Aircraft, Vol. 41, No. 2, pages 372-379

Howitt, Jeremy, 1995, "Carefree Handling for super-agility", American Helicopter Society 2nd Aeromechanics Specialist Conference, Fairfield County, Oct. 11-13 1995, pages 4-58

Jategaonkar, Ravindras V., 2006, *Flight vehicle system identification: A time domain Methodology*, AIAA Progress in Aeronautics and Astronautics, Arlington: Frank K. Lu, 534 pp.

Jeram, Geoffrey J., 2002, "Open design for helicopter active control system", American Helicopter Society 58th Annual Forum, Montreal, Canada, June 11-13 2002.

Jeram, Geoffrey J., Nilesh H. Sahani., J.V.R Prasad, 2005, "Distributing Limit Protection between Autonomous Restraint and Voluntary Tactile Cues ", 43th

AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 10-13. 2005, pp.1-14

Kalman, R.E., 1960, "A new approach to linear filtering and prediction problems", ASME Journal of Basic Engineering, Series 82D, Mar. 1960, pp. 35-45.

Ljung. Lennart , 1999, *System Identification Theory for the user*, Prentice Hall, Upper Saddle River, N.J., 2nd edition, pp.340-351

Ljung Lennart, 2006, *System Identification Toolbox for use with Matlab*®, User Guide Version 6, The Mathworks Inc., 416 pp.

Massey, C.P., and Wells P., 1988, "Helicopter carefree handling systems", Royal Aeronautical Society Conference on Helicopter Handling Qualities and Control, London, UK, 15-17 November 1988

McCool, Kelly, Lance A Flitter, and David J. Hass, 1998, "Development and Flight Test Evaluation of a Rotor System Load Monitoring Technology", American Helicopter Society 54th Annual Forum, Washington, 20-22 May 1998, pp. 408-418

Menon P.K., V. R. Iragavarapu, M.S. Walley 1996, "Estimation of Rotorcraft Limit Envelope Using Neural Networks", Annual forum proceedings of American Helicopter Society, Alexandria, Virginia, vol. 2, pp. 1423-1431

Rugh, Wilson J., 1993 , *Linear system theory*, Prentice Hall Information and System Sciences Series, New Jersey, 356 pp.

Sahasrabudhe Vineet, Robert Spauling, Alexandre Faynberg, Joseph Horn and Nilesh Sahani, 2002, "Simulation Investigation of a Comprehensive Collective-Axis Tactile Cueing System", in American Helicopter Society 58th annual forum, Montreal June 11-13 2002, pp. 559-568

Sahani, Nilesh A., December 2005, "*Envelope Protection Systems for Piloted and Unmanned Rotorcraft*", PhD Thesis, University park, Pennsylvania State University, 143 p.

Yavrucuk, Ilkay, J. V. R. Prasad and Antony J. Calise, 2001, "Adaptive Limit Detection and Avoidance for Carefree Maneuvering", AIAA Atmospheric Flight Mechanics Conference and Exhibit, Montreal, Aug. 6-9, 2001, pp. 1-9

Yavrucuk, Ilkay., J.V.R. Prasad and Antony J. Calise, 2002, "Carefree Maneuvering Using Adaptive Neural Networks", AIAA Atmospheric Flight Mechanics Conference abd Exhibit, Monterey, CA, Aug. 5-8 2002, pp. 291-299

Yavrucuk, Ilkay, J. V. R. Prasad, 2002, "Adaptive Limit Margin Prediction and Control Cueing for Carefree Maneuvering of VTOL Aircraft", American Helicopter Society Flight Controls and Crew System Design Technical Specialist Meeting, Philadelphia, Oct. 9-11, 2002, pp. 1-7

CHAPTER 7

INTRODUCTION TO THE THIRD PAPER

In this paper, the subspace system identification method was used, but, this time, to generate linear models to identify the structural deflections on different surfaces of the F/A-18 aircraft. During the flight tests, which took place at NASA DFRC laboratories, differential ailerons Schroeder frequency sweep control input was performed to excite the aircraft elastic modes of oscillations. The flight flutter tests used to generate the model were performed in steady level flight with Schroeder frequency excitations induced on the aircraft ailerons by an on-board excitation system (OBES) which was activated by the pilot. The F/A-18 linear model was conceived as nine Multiple Inputs Single Output models of third order. Each model of its own had nine inputs and one output. The nine inputs are the differential ailerons deflections and the deflections of all the other aircraft structural surfaces. The output of each model was the structural deflection of a given aircraft structure.

Published in the following journal:

1-Nadeau Beaulieu M., De Jesus Mota, S., Botez R. M., 2007, *Identification of structural surfaces positions of an F/A-18 using the subspace identification method from flight flutter tests*, Journal of Aerospace Engineering, Vol. 221 (5), pp. 719-731.

Presented to the following conferences:

1- Nadeau Beaulieu M., De Jesus Mota, S., Botez R. M., 2007, *Identification of structural surfaces positions of an F/A-18 using the subspace identification method from flight flutter tests*, 17th IFAC Symposium on Automatic Control in Aerospace, Toulouse, France, June 25-29 2007.

CHAPTER 8

IDENTIFICATION OF STRUCTURAL SURFACES POSITIONS ON AN F/A-18 FROM FLIGHT FLUTTER TEST USING THE SUBSPACE IDENTIFICATION METHOD

by

Michel Nadeau Beaulieu, Sandrine De Jesus Mota, Ruxandra Mihaela Botez
École de technologie supérieure
1100 rue Notre-Dame ouest
Montréal, Québec, Canada, H3C 1K3

8.1 Abstract

In this paper, we present a linear state-space mathematical model, identified from flight flutter tests to simulate the aeroelastic deflections of specific structural parts of the NASA F/A-18 aircraft. The flight flutter tests were performed in steady level flight with Schroeder frequency excitation induced on the aircraft ailerons by an on-board excitation system (OBES) which was activated by the pilot. We used the results of the flight flutter tests to generate an aeroelastic model in which the deflections of the specific aircraft surfaces are functions of the control inputs combined with the deflections of other aircraft surfaces. The F/A-18 linear model is conceived as nine Multiple Inputs Single Output models of third order. Each model of its own has nine inputs and one output. The nine inputs are the differential ailerons deflection and the deflections of all the other parts of the aircraft. The output of each model is the structural deflection of a given aircraft structure. The model's parameters are estimated with the subspace system identification algorithm, an efficient non-iterative algorithm that computes the system matrices directly from the inputs and outputs data. The model's quality is evaluated by calculating the fit and correlation coefficients between the model's outputs and the outputs from flight flutter test data. While the fit coefficient results are very good, between 89% and 99%, the correlation coefficient method gave the best results (nearly 100%). Finally, re-sampled inputs were used to validate the F/A-18 model robustness. The model's aircraft structure was validated for flutter flight tests at

different Mach numbers and altitudes. The estimated linear model fits very well the flight flutter tests data. The subspace method is therefore very convenient for model identification from flight flutter tests.

8.2 Introduction

This project uses flight flutter test data obtained from the F/A-18 aircraft. The details of the modifications performed on the F/A-18 aircraft in the Active Aeroelastic Wing AAW Flight Research Program were explained by Lind et al. (1999) and Voracek et al. (2003).

This paper determines a model for the structural oscillations of the flexible F/A-18. The model is build by use of the subspace parameter estimation methods from flight flutter tests. The next sections present a literature review on aeroelasticity where system identification methods and mainly the subspace method are used for model identification from flight flutter tests.

The Autoregressive Moving Average Method (ARMA) and neural networks theory were used by Sung (2005) to identify the flutter behaviour of a transonic wing. The flutter dynamics of a pitch-plunge system subjected to limit cycle oscillations was later modeled by Kukreja (2006) with non-linear models. In previous paper, the type of non-linear models used was the Nonlinear Autoregressive Moving Average Exogenous (NAMAX). The dynamics of a flexible wing model has also been identified by Silva and Vartio (2006) using the impulse response method and the Eigensystem Realisation Algorithm (ERA). Also, an output-error minimisation method was performed based on a large flexible aircraft by LeGarrec (2001).

The subspace method has already been applied in other fields such as fibre optics research by Galvao (2005). In Aerospace, the subspace method has been used to identify the effects of the aircraft's control surfaces motion on the rigid modes of the F/A-18 from flight flutter tests by Brenner (1997). In this previous method, the ailerons were excited by the use of

Schroeder frequency sweeps. The accelerations of the aircraft were filtered using a wavelet transform and the aircraft's roll response was identified in both time and frequency domains. In the present study, the subspace identification method is used to identify the structural deflections of the F/A-18 aircraft's surfaces from flight flutter tests.

8.3 Methodology

The methodology section of this paper is divided into three sections. The first section presents the flight flutter tests data for the F/A-18 aircraft. The second section presents the structure of the linear model with its inputs and outputs. The last section explains the details of the subspace identification algorithm that is used to obtain the parameters of the linear model.

8.3.1 Flight Flutter Tests Data and their Filtering

In order to obtain the recorded flight flutter tests data, the flight control computer (FCC) for the F/A-18 aircraft was modified by adding a Research Flight Control System (RFCS) to generate the Schroeder frequency sweep control inputs. The RFCS processor was engaged by the pilot from the activation of a cockpit switch. The actuator commands therefore resulted from the RFCS added to the commands from the aircraft's baseline flight control computer.

The software used by the RFCS to control the actuators was called the On Board Excitation System (OBES). The input activated by the OBES was a Schroeder frequency sweep, which is a large number of harmonics, equally spaced in frequency domain. An example of OBES controls is shown in figure 8.1.

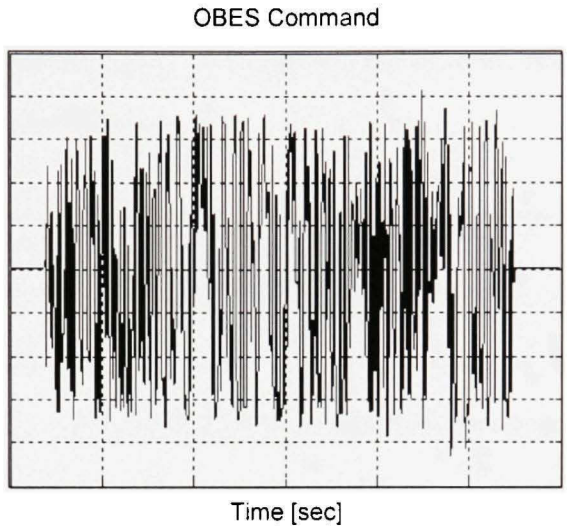

Figure 8.1 ***OBES control inputs versus time***

The OBES Schroeder excitation signal is defined in equation (8.1):

$$OBES(t) = \sum_{k=1}^{C} A_k \sin\left(2\pi f_k t + \phi_k\right) \tag{8.1}$$

where f_k is the k^{th} measurement frequency, ϕ_k is the k^{th} phase and A_k is the k^{th} amplitude of the OBES Schroeder signal. Details on the theory of Schroeder signals are given by Schroeder (1970). The OBES generated Schroeder signal is sent to the aircraft actuators to generate the F/A-18 control surface oscillations. Records of structural surfaces accelerations were obtained at 30-seconds intervals by accelerometers. These tests were performed for a combination of Mach numbers from 0.85 to 1.20 and for altitudes from 5000 ft to 25000 ft.

In this paper, we use firstly with the accelerations from flight flutter test data measured by NASA DRFC laboratories. The flight test data accelerations on the structural surfaces are very noisy. We remove the noise in order to identify the F/A-18 model by performing a double integration on the surface accelerations to obtain the surface deflections. Since the noise is a random process with a mean zero value, then the integration removes any noise contribution to the data. Therefore, no additional filter is required. Figure 8.2 shows the

schematic of the flight flutter test data pre-processing. The filtering effect of the acceleration double integration is shown in figure 8.2 where speeds and deflections time histories are presented for the left wing surface.

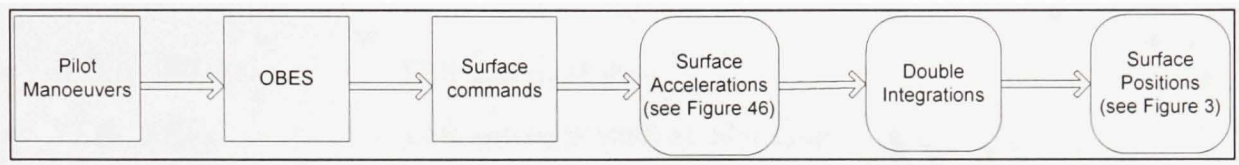

Figure 8.2 : ***Flight flutter tests data pre-processing scheme***

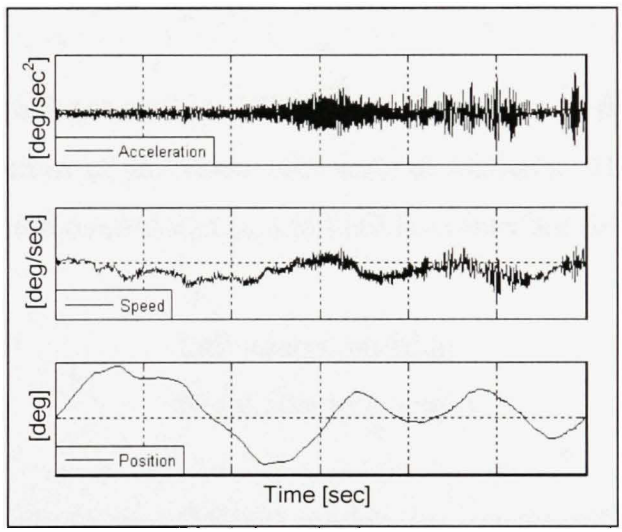

Figure 8.3 : ***Left wing accelerations and their integrations with time, which gives the deflection speed and the deflection with time***

In figure 8.3, only the structural surface deflection acceleration for the *left wing* is shown to illustrate the way in which the integrations operation removes the unwanted noise.

8.3.2

8.3.3 Linear Models

The deflections of the following nine structural surfaces are considered in this paper:

- $WING_L, WING_R$ Left and right wing
- TEF_L, TEF_R Left and right trailing edge flap
- $VERT_L$, $VERT_R$ Left and right rudder
- STB_L, STB_R Left and right stabilizer
- $LSTB_Y$ Left stabilizer lateral

We assume that each structural surface deflection is a function of the left and right ailerons inputs and of the deflection of the other nine aircraft surfaces. The right and left aileron positions are considered the control inputs, and their notations are below given:

- AIL_L Left aileron position
- AIL_R Right aileron position

Nine Multiple Input Single Output (MISO) models are considered in this study, therefore, there are nine estimated outputs. Figure 8.4 shows the MISO scheme, where in this case, the estimated output is the left wing deflection $WING_L$. Same type of scheme is applied to the other estimated outputs.

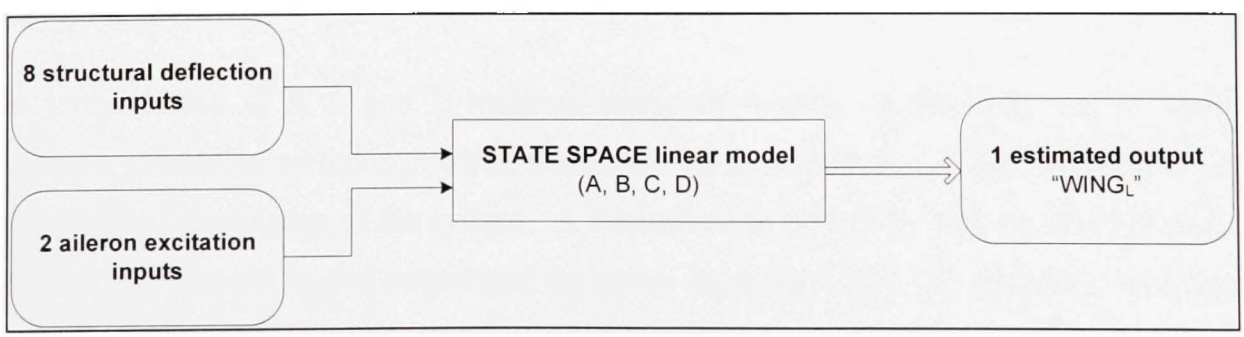

Figure 8.4 ***MISO model with the left wing output***

In this study, there are always 10 inputs associated to the MISO model: 9 structural surfaces (as above mentioned) minus one which is the estimated output (in this case the left wing) plus both right and left ailerons positions. The other 8 MISO models are used to obtain the remaining 8 structural surface deflections versus time in a similar manner as the one above explained.

8.3.4 Subspace system identification algorithm

The state space matrices parameters (figure 8.4) are obtained by the subspace system identification algorithm, which is described in this section. Generally, a discrete linear model is defined with the following equations (8.2) and (8.3):

$$x(t+\Delta T)_{n\times 1} = A_{n\times n}x(t)_{n\times 1} + B_{n\times m}u(t)_{m\times 1} + w(t)_{n\times 1} \tag{8.2}$$

$$y(t)_{o\times 1} = C_{o\times n}x(t)_{n\times 1} + D_{o\times m}u(t)_{m\times 1} + v(t)_{o\times 1} \tag{8.3}$$

where t is the time, ΔT is the sample time, $u(t)$ represents the model inputs, $y(t)$ represents the model outputs, m is the number of inputs and o is the number of outputs. In equations (8.2) and (8.3), the vector $x(t)$ of length n represents the system states expressed as linear combinations between previous inputs and previous outputs, $w(t)$ represents the state noise vector and $v(t)$ is the measurement noise vector. The matrices A, B, C and D are well known in the modern control theory.

The terms of the A, B, C and D matrices terms are usually estimated by use of various parameter estimation methods. Most of these methods start with a set of initial guesses found from physical knowledge of the system. A minimization algorithm is further used to reduce the error between the model output and the given flight test data. Unfortunately, with these methods, if the initial parameter guesses are far from their true values, the minimization algorithm may converge towards a local minimum. The main advantage of the subspace

identification algorithm used in this paper is that is non-iterative and does not require an initial guess of the terms in the matrices [*A, B, C, D*]. The only information required by the subspace method is the input and output data vectors.

The subspace identification algorithm is implemented with the Matlab System Identification Toolbox. The basic theory of this algorithm is described by Ljung (1999) and the manner in which the algorithm is implemented in Matlab® is given by Ljung (2006). The main concept of the subspace method is the definition of the system observability matrix Γ_r in equation (8.4) from modern control theory, where r represents a forward prediction horizon. This matrix can be obtained from the system inputs $u(t)$ and outputs $y(t)$ and its expression is the following:

$$\Gamma_r \overset{def}{=} \begin{bmatrix} C_{o\,x\,n} \\ CA_{o\,x\,n} \\ \cdots \\ CA^{r-1}{}_{o\,x\,n} \end{bmatrix}_{ro\,x\,n} \tag{8.4}$$

Once this observability matrix Γ_r is known, then the state space matrices [*A, B, C, D*] are obtained by the use of the input and output vectors.

The detailed procedure to obtain the observability matrix and the discrete state space matrices will now be explained. The theory behind the subspace algorithm is divided into four sections. Section (8.3.3.1) describes the basic matrices and equations necessary to the demonstration, sections (8.3.3.2) and (8.3.3.3) explain the two steps necessary to compute the observability matrix of equation (8.4) while the last section (8.3.3.4) explains the obtention of the discrete state space matrices *[A,B,C,D]* from the observability matrix. Finally, section (8.3.3.5) discusses the selection of an appropriate mathematical model order.

8.3.4.1 Basic definitions of input and output matrices

In order to understand the subspace algorithm, is necessary to define a number of important matrices. The input data given later than the reference time t can be arranged into a *Hankel* matrix as follows:

$$U_f \overset{def}{=} \begin{bmatrix} u[t]_{m\,x\,1} & u[t+\Delta t]_{m\,x\,1} & u[t+2\Delta t]_{m\,x\,1} & \cdots & u[t+(j-1)\Delta t]_{m\,x\,1} \\ u[t+\Delta t]_{m\,x\,1} & u[t+2\Delta t]_{m\,x\,1} & u[t+3\Delta t]_{m\,x\,1} & \cdots & u[t+j\Delta t]_{m\,x\,1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ u[t+(r-1)\Delta t]_{m\,x\,1} & u[t+r\Delta t]_{m\,x\,1} & u[t+(r+1)\Delta t]_{m\,x\,1} & \cdots & u[t+(r+j-2)\Delta t]_{m\,x\,1} \end{bmatrix}_{m\,x\,N} \tag{8.5}$$

In this matrix, the subscript f stands for *future* inputs due to the fact that only the inputs given later than at time t are included into the matrix. The subscript N represents the length of the output vector from flight flutter test and the index j is a dummy variable adjusted such that all data available in the identification are included on each line of the U_f matrix. Please note that if the expression in the brackets of the matrix element u has a value greater than N, the value of U_f is zero. Similar *Hankel* matrices can be constructed with the output vectors and defined as Y_f. The future state vector can be defined by the use of the following matrix:

$$X_f = \begin{bmatrix} x[t]_{n\,x\,1} & x[t+\Delta t]_{n\,x\,1} & \cdots & x[t+(j-1)\Delta t]_{n\,x\,1} \end{bmatrix}_{n\,x\,N} \tag{8.6}$$

Is also necessary to define the extended controllability matrix:

$$\Delta_r = \begin{bmatrix} \left(A^{r-1}B\right)_{n\,x\,1} & \left(A^{r-2}B\right)_{n\,x\,1} & \cdots & (AB)_{n\,x\,1} & B_{n\,x\,1} \end{bmatrix}_{n\,x\,r} \tag{8.7}$$

and finally the impulse response matrix is given by:

$$H_r = \begin{bmatrix} D_{o\,x\,m} & 0 & 0 & 0 & 0 \\ CB_{o\,x\,m} & D_{o\,x\,m} & 0 & \dots & 0 \\ CAB_{o\,x\,m} & CB_{o\,x\,m} & D_{o\,x\,m} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ CA^{r-2}B_{o\,x\,m} & CA^{r-3}B_{o\,x\,m} & CA^{r-4}B_{o\,x\,m} & \dots & D_{o\,x\,m} \end{bmatrix}_{ro\,x\,Nm} \tag{8.8}$$

It is also important to describe the noise effect of noise on the *future* output of the system. This noise effect is defined by use of the *Hankel* matrix V:

$$V \overset{def}{=} \begin{bmatrix} V_1[t]_{o\,x\,1} & V_1[t+\Delta t]_{o\,x\,1} & V_1[t+2\Delta t]_{o\,x\,1} & \dots & V_1[t+(j-1)\Delta t]_{o\,x\,1} \\ V_2[t+\Delta t]_{o\,x\,1} & V_2[t+2\Delta t]_{o\,x\,1} & V_2[t+3\Delta t]_{o\,x\,1} & \dots & V_2[t+j\Delta t]_{o\,x\,1} \\ \dots & \dots & \dots & \dots & \dots \\ V_k[t+(r-1)\Delta t]_{o\,x\,1} & V_k[t+r\Delta t]_{o\,x\,1} & V_k[t+(r+1)\Delta t]_{o\,x\,1} & \dots & V_k[t+(r+j-2)\Delta t]_{o\,x\,1} \end{bmatrix}_{ro\,x\,N} \tag{8.9}$$

where the value of the term V_k is defined with equation (8.10):

$$V_k = CA^{k-2}w(t) + CA^{k-3}w(t+1) + \dots + Cw(t+k-2) + v(t+k-1) \tag{8.10}$$

In the subspace algorithm, the *future* value of the output is related to the future value of the states and input by use of the following equations:

$$X_f = A^r X_p + \Delta_r U_p \tag{8.11.1}$$

$$Y_f = \Gamma_r X_f + H_r U_f + V \tag{8.11.2}$$

Equations (8.11.1 and 8.11.2) were then derived from state-space equations (8.2) and (8.3). The next section will show how the observability term $\Gamma_r X_f$ can be isolated from equation (8.11.2).

8.3.4.2 Removal of input and noise contribution to the outputs

The first step to obtain the observability matrix Γ_r is to isolate the term dependant on the states $\Gamma_r X_f$ in equation (8.11.2). This procedure can be divided into two steps: 1. an orthogonal projection done to remove the input contribution $H_r U_f$ and 2. An instrument variable done to remove the noise contribution V. The algorithm used to perform this procedure is called *instrumental variables approach* and is thoroughly explained in references by Viberg et al. (1995, 1997) and Ljung (1999), while here is shortly described in following sections A) and B).

A) Orthogonal projection to remove the input contribution

To remove the input contribution $H_r U_f$, first a geometric interpretation of equation (8.11.2) needs to be used as shown in figure 8.5 and also explained in by Galvao (2005).

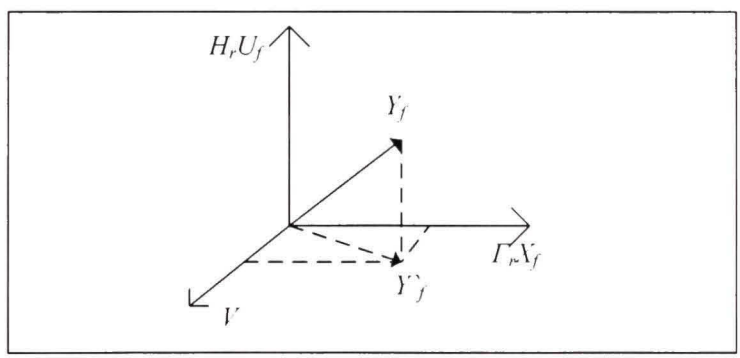

Figure 8.5 ***Perpendicular projection of the future outputs perpendicular to the future inputs***

If one interprets equation (8.11.2) as a vector, the output contribution can be removed by projecting the output vector Y_f perpendicular to the input contribution $H_r U_f$ which can be achieved with the following projection matrix:

$$\left[\Pi_{U^T}^{\perp}\right]_{N \times N} = I - U_f^T \left(U_f U_f^T\right)^{-1} U_f \tag{8.12}$$

where the superscript T means *transpose*. A detailed proof concerning this orthogonal projection matrix is available in Galvao's paper (2005). If this perpendicular projection operator is applied on U_f, it is equivalent to find the projection of $H_r U_f$ perpendicular to U_f which, logically, is zero. Mathematically, it can be shown by use of equation (8.13):

$$U_f \Pi_{U_f^T}^{\perp} = U_f - U_f U_f^T \left(U_f U_f^T \right)^{-1} U_f = U_f - U_f I = 0 \tag{8.13}$$

Post-multiplying both sides of equation (8.11.2) by the projection operator $\Pi_{U_f^T}^{\perp}$ yields:

$$\left[Y_f \Pi_{U_f^T}^{\perp} \right]_{r \times N} = \Gamma_r X_f \Pi_{U_f^T}^{\perp} + H_r U_f \Pi_{U_f^T}^{\perp} + V \Pi_{U_f^T}^{\perp} = \left[\Gamma_r X_f \Pi_{U_f^T}^{\perp} \right]_{r \times N} + \left[V \Pi_{U_f^T}^{\perp} \right]_{r \times N} \tag{8.14}$$

The input contribution has now been removed from the prediction equation.

B) Instrument variable to remove the noise contribution

It is now necessary to exclude the noise term $V \Pi_{U_f^T}^{\perp}$ of equation (8.14). This can be done by post-multiplying equation (8.14) with a suitable matrix Φ that is not correlated to the noise matrix V. Let's define the matrix Φ as follows:

$$\Phi = [\varphi_s(t) \quad \varphi_s(t+1) \quad \ldots \quad \varphi_s(t+j-1)]_{s \times N} \tag{8.15}$$

where φ_s are vectors that are uncorrelated with the noise. The number of lines s of this matrix may have any value, but it has to be higher than the desired order n of the dynamic system. This variable is called an *instrument variable* as defined by Viberg (1997) and Ljung (1999) and it is used to reduce the noise term of equation (8.14) to zero. Let's post-multiply equation (8.14) by the transpose of Φ and normalize with the number of sample (by dividing the equation by N) in the data as follows:

$$[G]_{ro\times s} \overset{def}{=} \frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T + \frac{1}{N} V \Pi_{U^T}^{\perp} \Phi^T \overset{def}{=} [\Gamma_r T_N]_{ro\times s} + [F_N]_{ro\times s} \tag{8.16}$$

In equation (8.16), the subscript N implies that the value of T_N and F_N are approximated for a data record containing N data points. In order to cancel out the noise term without affecting the term dependant on the future states T_N, the requirement for a proper *instrument* matrix Φ is that it must be correlated to the future states X_f, but uncorrelated with the noise term, mathematically, it can be expressed by the following equations:

$$\lim_{N\to\infty} F_N = \lim_{N\to\infty} \frac{1}{N} V \Pi_{U^T}^{\perp} \Phi^T = 0 \tag{8.17}$$

$$\lim_{N\to\infty} T_N = \lim_{N\to\infty} \frac{1}{N} X_f \Pi_{U^T}^{\perp} \Phi^T = T \tag{8.18}$$

In equation (8.18), the parameter T is equivalent to the estimation of T_N with an infinite number of data points. Equation (8.17) implies that as the number of samples N goes to infinity, the noise matrix and the *instrument matrix* Φ must cancel each other and F_N goes to zero. The equation (8.16) can therefore be summarized as:

$$G = \frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T \approx \Gamma_r T \tag{8.19}$$

The remaining step of the above demonstration is to find an appropriate *instrument* matrix Φ. As it was stated before, the first requirement for the *instrument* matrix Φ is that it must be correlated with the future states, which are unknown at this point. Even though these states are unknown, their value is dependant on the past inputs and outputs, these past inputs and outputs are therefore a possible choice. The second requirement is that the matrix Φ must be uncorrelated with the noise. This is always the case when the system inputs are properly selected. This is because the noise can be seen as the error between the model and the data and, for a good model, this error is completely random. The past input and outputs are

therefore a logical choice for the *instrument* matrix. In this paper, the *instrument* matrix used is described in equation (8.20) which was taken from Ljung (1999).

$$\Phi = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}_{s \times N} = \begin{bmatrix} y[t-1]_{o \times 1} & y[t]_{o \times 1} & \cdots & y[t+j-2]_{o \times 1} \\ \cdots & \cdots & \cdots & \cdots \\ y[t-h]_{o \times 1} & y[t-h+1]_{o \times 1} & \cdots & y[t-h+j-1]_{o \times 1} \\ - & - & - & - \\ u[t-1]_{m \times 1} & u[t]_{m \times 1} & \cdots & u[t+j-2]_{m \times 1} \\ \cdots & \cdots & \cdots & \cdots \\ u[t-h]_{m \times 1} & u[t-h+1]_{m \times 1} & \cdots & u[t-h+j-1]_{m \times 1} \end{bmatrix}_{s \times N} \tag{8.20}$$

In this equation, the subscript p stands for *past* input and outputs and the parameter h is the number of past inputs and outputs used by the algorithm. Once a proper instrument matrix has been found, the output equation can be described with equation (8.19). Combining equations (8.16) and equation (8.19) yields:

$$\frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T \tag{8.21}$$

At this point, every term of the left hand side of equation (8.21) are known. The next step will be to extract the observability matrix Γ_r. This can be done by performing a Singular Value Decomposition (*SDV*). More details on this procedure will be explained in section 8.3.3.3.

8.3.4.3 Determination of the Observability matrix from Singular Value Decomposition

Once the matrix G is known from equation (8.19), it is possible to decompose it into three sub-matrices using a well known linear algebra theorem called *Singular Value Decomposition*. This theorem is explained in through details in reference by Patel et al. (1993). This decomposition is expressed as follows:

$$G_{ro \times s} = U_{ro \times ro} S_{ro \times s} V^T_{s \times s} \tag{8.22}$$

In equation (8.22), the matrix S is made of the singular values of G. These singular values are the positive square root of the eigenvalues of $\left(G^T G\right)$, these eigenvalues are sorted in descending order from the first row to the last row of matrix S. The matrices U and V are called singular vectors. They are respectively the orthonormal eigenvectors of $\left(GG^T\right)$ and of $\left(G^T G\right)$. The following demonstration will show how the new matrices defined by the *Singular Value Decomposition* can be used to obtain an extended observability matrix Γ_r that relates properly the inputs to the outputs. The singular value decomposition provides a possible combination of matrices whose product gives the matrix G. Of these two matrices, the first one is a possible observability matrix and the second one is a possible state vector. The singular value decomposition of equation (8.22) applies for a dynamic system of any order. If the desired order n of the dynamic system is known, a proper procedure separate the first n singular values and singular vectors of the system from the other singular values and vectors. In practice, when performing singular value decomposition, only the system significant singular value and their corresponding singular vectors must be kept. The number of singular values considered corresponds to the assumed order of the model. The singular values that are kept should represent the true dynamic of the system and the small singular values that are not taken into account should correspond to errors due to noise. Once the singular value decomposition is done, it is possible to obtain a proper estimate of the observability matrix Γ_r. Notice that many different combinations of observability matrix and T matrix can lead to a set of parameters value that insures a proper match. According to reference Ljung (1999), it follows that the value of the observability matrix may be expressed with the following equation:

$$\Gamma_r = U \tag{8.23}$$

It is also possible to add weight functions to the matrix G obtained from equation (8.23) before performing singular value decomposition as follows:

$$G_{Weight} = W_1 G W_2 \tag{8.24}$$

The reason for adding weight matrices is to remove any residual error due to noise in the variable G. These weight matrices are made of parameters that are uncorrelated with the projected noise matrix F_N. In the absence of noise, adding a weight matrix has no effect on the identification results, but it improves the results when the identification is done on noisy data.

After a weight matrix is selected, the new observability matrix is found by use of the following equation:

$$\Gamma_r = W_1^{-1} U \tag{8.25}$$

Many authors have proposed expressions for different weight matrices and they have an influence on the results of the identification. A good summary on the work of different authors on these weight matrices formulations has been done by Viberg (1997) where different expressions for weight matrices are derived using the same mathematical approach. For original work on the different weight matrices, the reader is invited to consult the papers written by Verhaegen (1994), Van Overschee (1994), Viberg (1995) and Larimore (1990). In this paper, the weight matrices defined by Larimore (1990) were tried and gave excellent results. Since there is not much noise in the data related to this project, the algorithm is not very sensitive to the selected weight and it wasn't necessary to try other weight formulations to obtain good results. The weights as defined by Larimore are defined by use of the following equation:

$$W_1 = \left(\frac{1}{N} Y \Pi^{\perp}_{U^T} Y^T \right)^{-1/2} ; W_2 = \left(\frac{1}{N} \Phi \Pi^{\perp}_{U^T} \Phi^T \right)^{-1/2} \tag{8.26}$$

At this point the observability matrix Γ_r have been determined from equation (8.25). This observability matrix can now be used to obtain the value of the matrices [A, B, C, D]. The procedure to do it will be shown in the next section.

8.3.4.4 Determination of the system matrices [A,B,C,D] from the observability matrix

A) Estimating A and C

Once the observability matrix is known, it is quite easy to obtain the estimates of the A and C matrices. If we refer to equation (8.4) as it was defined at the beginning of the theory section, the estimate matrix C is obtained by taking the following terms of the observability matrix:

$$\hat{C} = \Gamma_{(1:o,1:n)} \tag{8.27}$$

Where the hat "^" means that it is an estimate. Equation (8.27) simply means that we have to extract the first o lines and n columns of the observability matrix. The matrix A can be found from the observability matrix by solving the following equation:

$$\Gamma_{(o+1:ro,1:n)} = \Gamma_{(1:o(r-1),1:n)} A \tag{8.28}$$

In this equation, the left hand side represents the observability matrix with the first sub-matrix $\hat{C}$ removed and the right hand side represents the observability matrix with the last sub-matrix CA^{r-1} removed. This is equivalent to the following equation:

$$\begin{bmatrix} CA_{o\,x\,n} \\ CA^{2}{}_{o\,x\,n} \\ \cdots \\ CA^{r-1}{}_{o\,x\,n} \end{bmatrix}_{(r-1)o\,x\,n} = \begin{bmatrix} C_{o\,x\,n} \\ CA_{o\,x\,n} \\ \cdots \\ CA^{r-2}{}_{o\,x\,n} \end{bmatrix}_{(r-1)o\,x\,n} A_{n\,x\,n} \tag{8.29}$$

In this equation, the only unknown is the state matrix A. If the modified observability matrices $\Gamma_{(o+1:ro,1:n)}$ and $\Gamma_{(1:o(r-1),1:n)}$ of equation (8.29) were both square matrices, it would be easy to find $\hat{A}_{n\,x\,n}$ by just pre-multiplying both side of the equation by the inverse of $\Gamma_{(1:o(r-1),1:n)}$. Since these matrices are not necessarily square, equation (8.29) can be solved by use of the following equation:

$$\hat{A}_{n\,x\,n} = \left(\begin{bmatrix} C_{o\,x\,n} \\ CA_{o\,x\,n} \\ \cdots \\ CA^{r-2}{}_{o\,x\,n} \end{bmatrix}^{\dagger} \right)_{n\,x\,(r-1)o} \begin{bmatrix} CA_{o\,x\,n} \\ CA^{2}{}_{o\,x\,n} \\ \cdots \\ CA^{r-1}{}_{o\,x\,n} \end{bmatrix}_{(r-1)o\,x\,n} \tag{8.30}$$

In equation (8.30), the superscript "$\dagger$" denotes the Moore-Penrose pseudo-inverse described by Viberg (1995). This is a more general type of inversion which does not require the matrices to be square. The pseudo-inverse can be computed by singular value decomposition and reader is invited to consult references from Patel (1993) and Klema (1980) for more details on this operation.

B) Estimating *B* and *D*

Once the system matrices $\hat{A}$ and $\hat{C}$ are known, it is possible to estimate the B and D matrices by use of a linear regression technique. The discrete state-space model of equation (8.2) and (8.3) can be converted into a discrete transfer function by using the discrete operator z defined with the following equation:

$$\begin{aligned} zx(t) &= x(t + \Delta t) \\ z^{-1}x(t) &= x(t - \Delta t) \end{aligned} \tag{8.31}$$

which leads to the following transfer function:

$$\hat{y}(t \mid B, D) = \hat{C}\left(zI - \hat{A}\right)^{-1} Bu(t) + Du(t) \tag{8.32}$$

where $\hat{y}(t \mid B, D)$ means the *estimated* output which depends on the values of the matrices B and D. A very efficient way to find the unknown parameters B and D of equation (8.32) is to use a linear regression method. The estimated output $\hat{y}(t)$ may be expressed by the following equation:

$$\hat{y}(t)_{o \times 1} = \eta(t)\hat{\theta} = \eta(t)_{o \times (nm+om)} \begin{bmatrix} Vec(B) \\ Vec(D) \end{bmatrix}_{(nm+om) \times 1} \tag{8.33}$$

In equation (8.33), the matrix $\eta(t)$ is made of the past and presents inputs $u(t)$. The single column vector $\hat{\theta}$ represents all the estimated parameters to be found by the regression, these unknown parameters are all the elements of the matrices B and D. The operator "Vec" builds a column vector from a matrix by stacking its columns on top of each other.

8.3.4.5 Selection of the mathematical model order

The aircraft model structure should be defined with enough number of parameters to obtain good results, but should be parsimonious due to the fact that a model with too many parameters may overfit the data. The Akaike (1969) information criterion was used to compute a modified cost function J which takes into account the number of parameters d_M used to define the model. This criterion is defined as follows:

$$J = \frac{1 + d_M}{1 - d_M} \left\{ \frac{1}{N} \sum_{t=1}^{N} \left[y(t) - \hat{y}(t) \right]^2 \right\} \tag{8.34}$$

Where the cost function J decreases when the mean square error between the model and the data decreases, but increases with the number of model parameters d_M. The best model is the optimal compromise between a low mean square error and a model with a small number of parameters and will actually have the lowest possible cost function. In Table 8.1, the cost functions J are calculated with equation (8.34) for the second, third and fourth order models for all nine MISO models. The minimum value of J is found with the third model order which is therefore chosen to be the optimal model order.

Table 8.1

Model order selection based on cost function J values

Model order	**Order 2**	**Order 3**	**Order 4**
Cost function J	6.2591×10^{-6}	2.2699×10^{-6}	3.9055×10^{-5}

8.4 Results

This section of this paper is divided into the following three parts:

3.1 An example of the goodness of the match between the model output and the flight test data is shown graphically for the flight condition characterized by the Mach number = 0.85 and the altitude of 5,000 ft.

3.2 The criterions used to evaluate the model for all flight conditions are explained.

3.3 The results for every flight conditions in terms of these criterions are explained.

8.4.1 Results for the flight condition characterized by one Mach number = 0.85 and altitude of 5,000 feet

Figure 8.6 shows the nine structural deflections on different parts of the aircraft with respect to time. The results presented in figure 8.6 are given for a flight test condition at 5,000 ft at a constant Mach number of 0.85. The model's estimation results are represented by full lines, and the stars represent the flight flutter tests data.

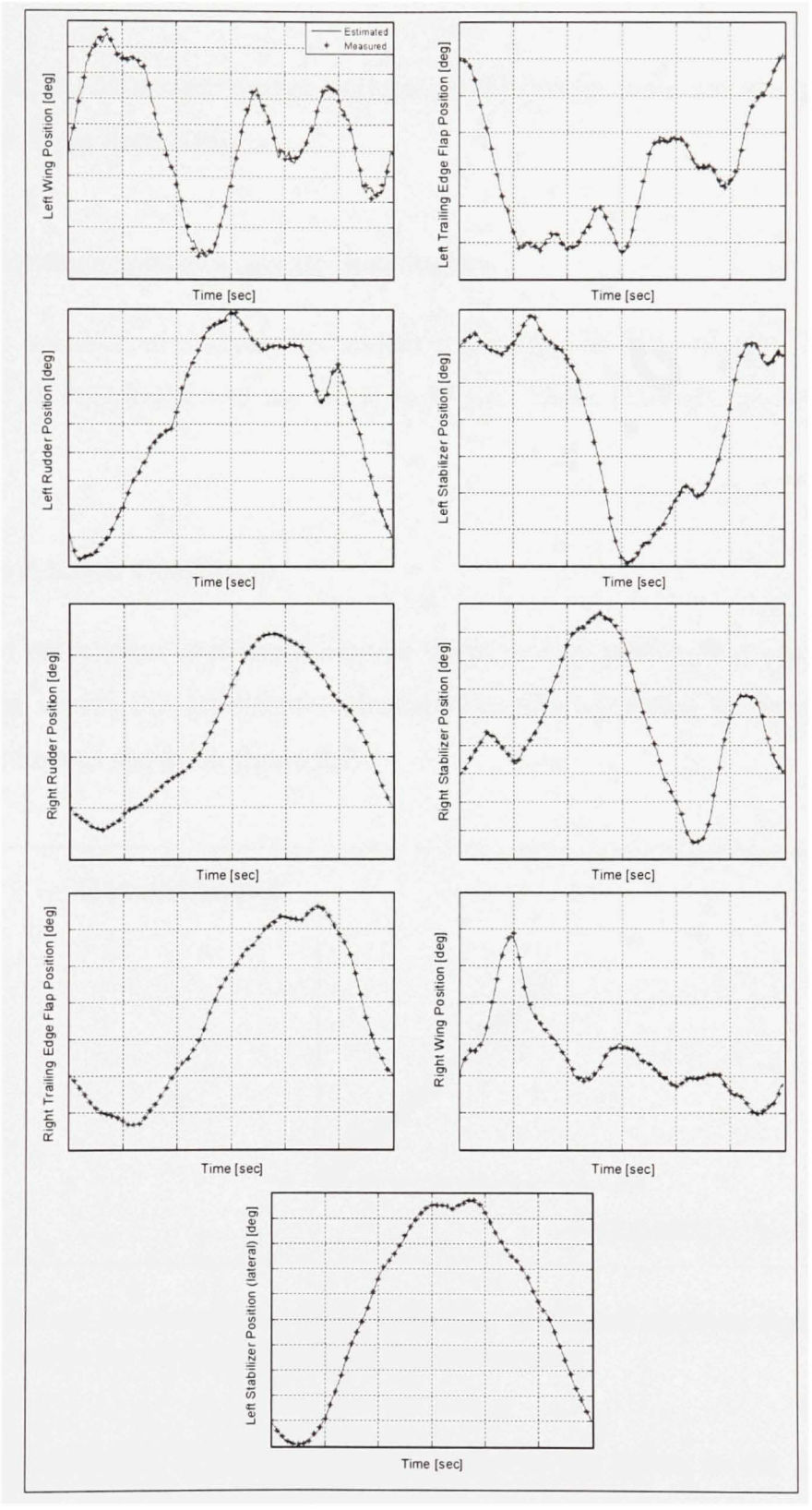

Figure 8.6 ***State-space model estimation of structural surface deflections (full line) and their measurements (or flight tests data) with respect to time.***

From figure 8.6, the estimated surface deflections follow the same dynamics as the surface deflection from flight flutter tests.

8.4.2 Criterions used to evaluate the results

Three methods are used to evaluate the model's accuracy of these results: The *correlation coefficient*, the *fit coefficient* and the *robustness test*. These methods are explained in this section.

8.4.2.1 Correlation Coefficient

The concept of correlation coefficient may be illustrated by plotting the results from flight flutter test data versus the estimated (calculated) results expressed in terms of structural surfaces deflections as shown in figure 8.7:

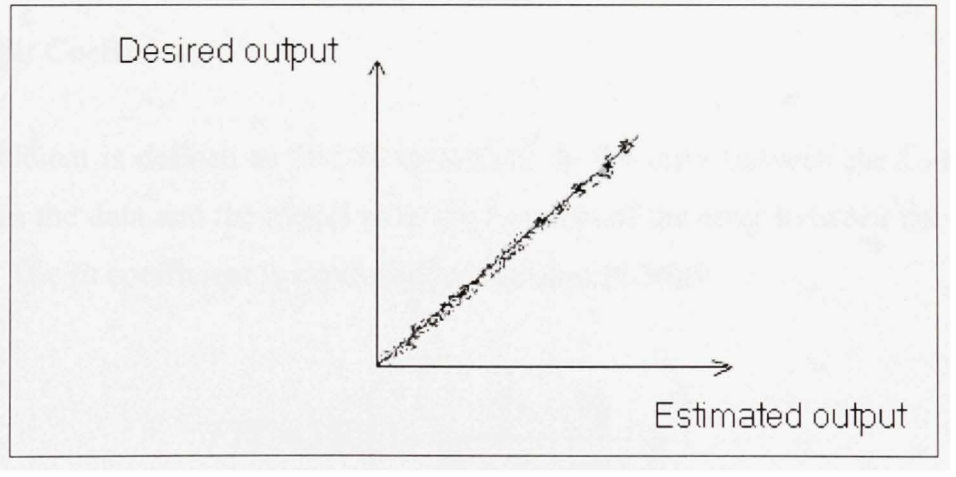

Figure 8.7 ***Visual interpretations of the estimated structural surfaces deflection outputs versus the output from flight flutter test data.***

The correlation coefficient R is defined as a measure of the scatter in the graph shown in figure 8.7 between the output from flight flutter test data and the calculated (estimated) output. Mathematically, its expression is given by the following equation:

$$R = \frac{Cov(y,\hat{y})}{\sqrt{Var(y)Var(\hat{y})}} \tag{8.35}$$

where Cov is the covariance, Var is the variance, y is the output from flight flutter test data and $\hat{y}$ is the estimated output. The value of the correlation coefficient R is situated between –1 and 1. The correlation coefficient R equal to one (R = 1) denotes *perfect linear dependency* (no scatter) between the outputs from flight flutter test data with respect to the calculated or estimated outputs. A correlation coefficient equal to minus one (R = -1) denotes *inverse linear dependency* between the estimated output and the output from flight flutter test data. A correlation coefficient of zero (R = 0) denotes the *linear independency* between the estimated outputs and the output from flight flutter test. The correlation coefficient computes the goodness of the model in a statistical sense, but provides little information about the model error. More information can be obtained by the second method: the *fit* coefficient.

8.4.2.2 Fit Coefficient

The fit coefficient is defined as 100 % multiplied by the ratio between the L_2-norm of the error between the data and the model over the L_2-norm of the error between the data and its mean value. The *fit* coefficient is expressed by equation (8.36):

$$FIT = 100\left(1 - \frac{\|y - \hat{y}\|}{\|y - mean(y)\|}\right) \tag{8.36}$$

Where the term in $(\|y - \hat{y}\|)$ is the L_2-norm of the error. The L_2-norm is defined with equation (8.37):

$$\|y-\hat{y}\| = \frac{1}{2}\sqrt{\sum_{1}^{n}(y-\hat{y})^2} \tag{8.37}$$

Equation (8.36) therefore becomes:

$$FIT = 100\left(1-\sqrt{\frac{\sum_{1}^{n}(y-\hat{y})^2}{\sum_{1}^{n}(y-mean(y))^2}}\right) \tag{8.38}$$

In equation (8.38), the term under the square root represents the ratio between two residuals: the residual between the model and the data and the residual between the data and their mean value. Intuitively, the fit coefficient explains the percentage of data variation that is explained by the model. The main advantage of using the *fit* coefficient is that it takes into account how the data varies about its mean in order to evaluate the model quality. For example, even if a model has an output very close to the data output, it will have a poor *fit* coefficient if there are much more small oscillations in the data than in the model.

8.4.3 Robustness test

In order to test our estimated model robustness from flight tests data, we consider re-sampled signals that use one point over one hundred (100). The signals are further re-sampled, as there is the need to use the same sampling rate as that used in the model implementation (T_s = 0.01), see figure 8.8.

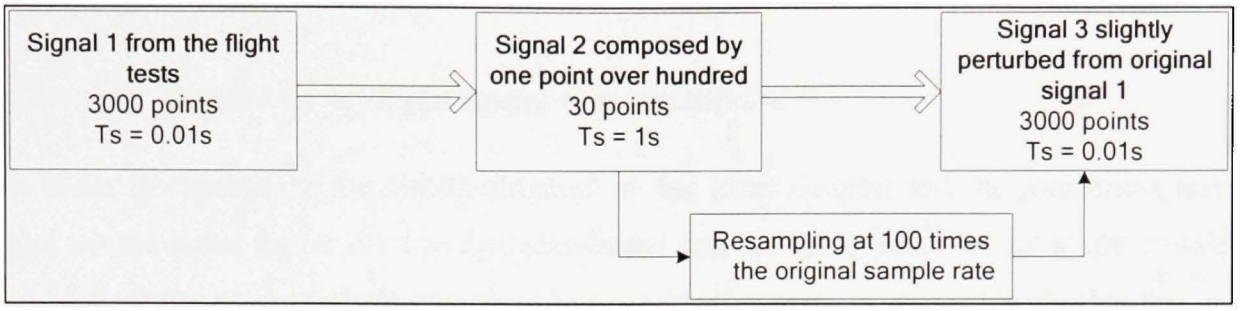

Figure 8.8 ***Robustness test methodology***

The schematic shown in figure 8.8 is equivalent to a procedure that makes perturbations on the model inputs and outputs in order to check its robustness. The reconstructed inputs are further used in the initial model. This is equivalent to adding many small perturbations to the inputs signals, in order to measure the sensitivity of the model to these perturbations. In figure 8.8 the initial signals from flight test is very close to the re-sampled signals.

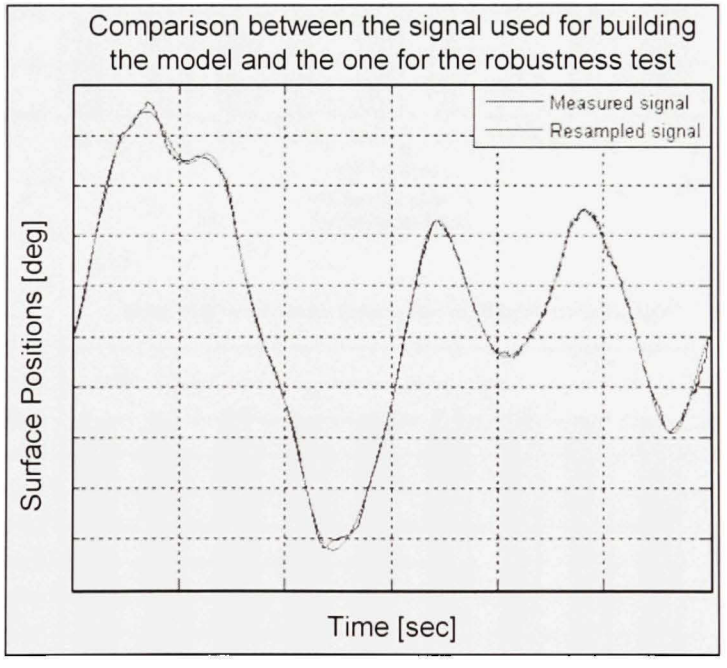

Figure 8.8: ***Initial flight flutter test data versus re-sampled signals***

If the model is robust, the model output from the re-sampled input should be similar to the model output when using the initial input.

8.4.4 Results for all flight flutter tests conditions

In order to summarize the results obtained by the identification and the robustness test, we plot on the same figure the average correlation and fit coefficients for all flight conditions and for each surface deflection in the upper and lower parts of figure 8.9. In this figure, the full bars represent the results by the identification and the dashed bars represent the results of the robustness test. The black lines represent the standard deviation of each coefficient for all flight conditions.

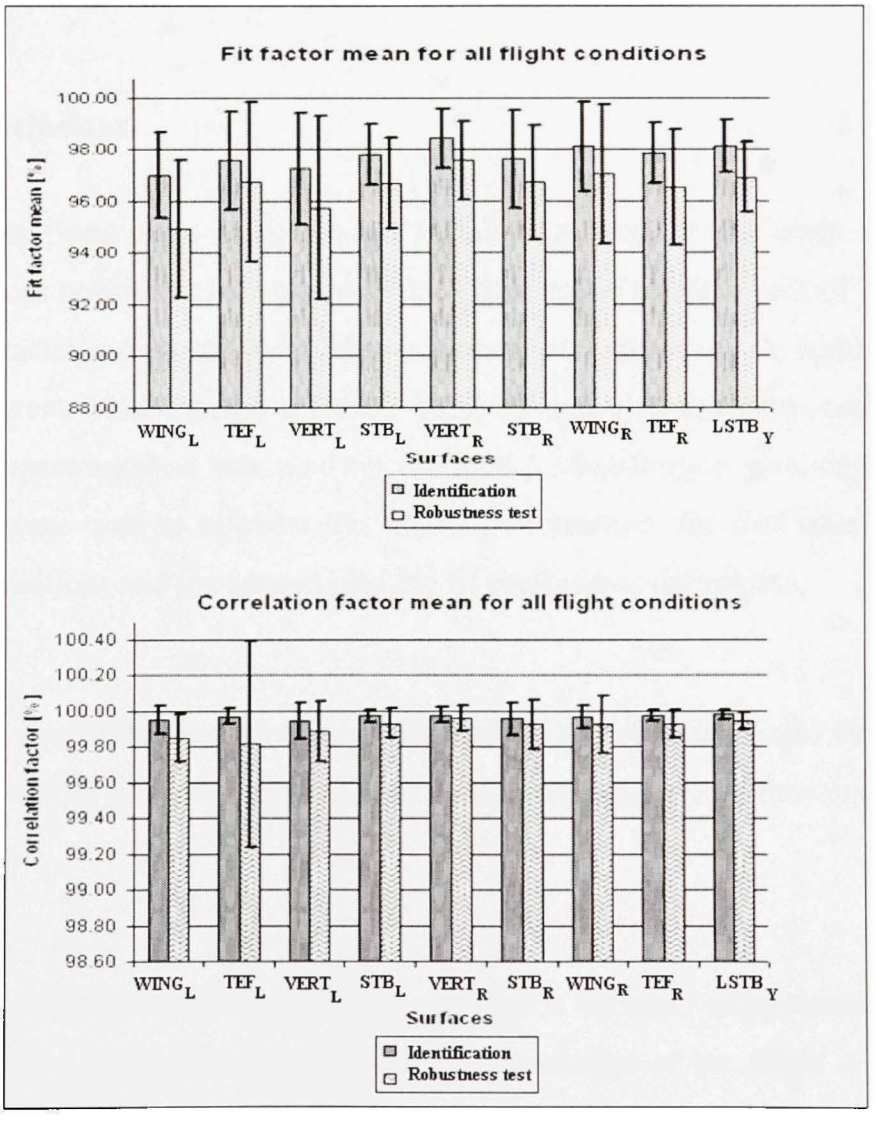

Figure 8.9 ***Correlation and fit factors mean and standard deviation for all flight conditions***

Firstly, we observe that the fit coefficients are very good for the identification part (model construction) due to the fact that the average fit coefficient is found to be higher than 97% for each surface, which shows that our calculated model validates the real aircraft system (from flight tests).

We can also note that for the robustness tests we obtained fit and correlation coefficients of almost the same magnitude as those obtained in the identification tests. The differences between the two sets of results are 1% to 2%. We therefore conclude that our estimated model is robust.

8.5 Conclusions

Nine third order linear state space models MISO were used in this study to estimate the structural surfaces positions given by the F/A-18 differential ailerons control inputs for flight conditions characterized by different Mach numbers and altitudes. A number of 19 flight tests with different combinations of Mach numbers and altitudes were considered in this study. The subspace method was used for the model identification from flight flutter tests. Two methods were used to estimate this model performance: the first uses the correlation coefficients definitions and the second uses the fit coefficients definitions.

The correlation coefficients are close to 100% and the fit coefficients also have high values, with the worse values of 89%. Therefore, the estimated linear model fits very well the flight flutter tests data.

The advantage of the subspace identification method is its small computation time and also the estimation of a very good model from the knowledge of the flight tests' inputs and outputs without a-priori knowledge about the model dynamics. The estimated model was found to be robust by the re-sampling technique application. We further conclude from the

obtained results that the subspace method is very convenient for model identification from flight flutter tests.

8.6 Acknowledgements

The authors would like to thank Mr. Marty Brenner at NASA Dryden Research Flight Center for his collaboration. Financial support for the work related to this paper was given by the Natural Sciences and Engineering Research Council of Canada NSERC and by the Ministère du Développement économique, de l'innovation et de l'exportation MDEII.

8.7 References

Akaike, Hirotsugu., 1969, *Fitting autoregressive models for prediction,* Ann. Inst. Stat. Math., Vol. 21, pp. 243-347

Brenner Martin, Eric Feron, April 1997, "Wavelet analysis of F/A-18 aeroelastic and aeroservoelastic flight test data", 38th AIAA/ASME/ASCE/AHS/ASC Structure, Structural Dynamics and Material Conference Exhibit, Kissimmee, Florida, 7-10 April, 1997., pp. 691-713

Galvao, Roberto K. H., Sillas Hadjiloucas, Victor M. Becerra and John W Bowen, "Subspace system identification framework for the analysis of multimoded propagation of THz-transient signals", Institute of Physics Publishing, Measurement Science and Technology, Vol. 16, pp. 1037-1053, 2005.

Klema Virginia C., Alan J. Laub, 1980, "The Singular Value Decomposition: Its Computation and Some Applications", IEEE Transaction on Automatic Control, vol. AC-25, no. 2, pp. 164-176

Kukreja, Sunil L., Martin Brenner M., "Nonlinear Aeroelastic system identification with application to experimental data", AIAA Journal of Guidance, Control and Dynamics, Vol. 29, No. 2, 2006.

Larimore, Wallace E., 1990, "Canonical variate Analysis in Identification, Filtering, and Adaptive Control" , In proc. 29th IEEE Conference on Decision and Control, Honolulu, Hawaii, December 1990

Le Garrec C., Humbert M., Bucharles A., Vacher P., "In flight aeroelastic model identification and tuning of a flight control system on a large civil aircraft", CEAS International Forum on Aeroelasticity and Structural Dynamics, Madrid, Spain, 5-7 June, 2001.

Lind, Rick, Marty Brenner, (1999), *Robust aeroservoelastic stability analysis: Flight Test Applications*, Advances in industrial control, Springer-Verlag Ed, 204 pp.

Ljung. Lennart , 1999, *System Identification Theory for the user*, Prentice Hall, Upper Saddle River, N.J., 2nd edition, pp.340-351

Ljung Lennart, 2006, *System Identification Toolbox for use with Matlab®*, User Guide Version 6, The Mathworks Inc., 416 pp.

Patel, RajniV., Alan J. Laub, Paul M. Van Dooren., 1993, *Numerical Linear Algebra Techniques for System and Control*, IEEE Press, City College of New York, 724 pp

Schroeder M.R., 1970, "Synthesis of low-peak factor signals and binary sequences with low autocorrelation," *IEEE Trans. Inform. Theory,* Vol. IT-16, Jan. 1970., pp. 85-89

Silva Walter A., Eric Vartio, Anthony Shimko, Raymond G. Kvaternik, Kenneth W. Eure and Robert C. Scott, Development of Aeroservoelastic Analytical Models and Gust

Load Alleviation Control Laws of a SensorCraft Wind-Tunnel Model Using Measured Data, 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Newport, USA, 4-6 May, 2006.

Sung Won Kok, Her Mann Tsai, Mani Sadeghi, Feng Liu, "Non-linear impulse methods for aeroelastic simulations", 23rd AIAA Applied Aerodynamics Conference, Toronto, Canada, 6-9 June 2005, pp. 1-19.

Van Overschee Peter, Bart De Moor, 1994, "N4SID: Subspace algorithm for the identification of combined deterministic-stochastic systems", Automatica, 30(1), pp. 75-93

Verhaegen, Michael , 1994, "Identification of the deterministic part of MIMO state space models, given in innovation form from input-output data", Automatica, 30(1), pp. 61-74

Viberg, Matt, 1995, "Subspace-based Methods for the identification of Linear Time-invariant Systems", Automatica, 31(12), pp. 1835-1852

Viberg, Matt., Bo Wahlberg, and Bjorn Ottersen, 1997, "Analysis of state space system identification methods based on instrumental variables and subspace fitting", Automatica 33(9) pp. 1603-1616

Voracek D., Pendleton E., Reichenback E., Griffin K., Welch L., 2003, *The active aeroelastic wing phase I Flight Research through January 2003*, NASA Dryden Flight Research Center, NASA/TM-2003-210741, 25 pp.

CHAPTER 9

INTRODUCTION TO THE FOURTH PAPER

The aim of this paper was to determine the mathematical relationship (model) between control deflections and structural deflections of the F/A-18 aircraft. This paper represented an improvement with respect to the third one. In the third paper, different MISO models were used to simulate the deflections of the different structural surfaces of the F/A-18 following a differential ailerons deflection. In this paper, a single Multiple Inputs Single Outputs (MIMO) model was used to estimate simultaneously the structural deflections of the four following surfaces: the left wing, the right wing, the left training edge flap and the right trailing edge flap.

The model was built using five different Schroeder excitations: differential ailerons, collective ailerons, collective stabilizers, differential stabilizers, and rudders. In the same manner as in the third paper, the mathematical model was represented by a state-space model, but, this time, non-lineal inputs were added to the linear inputs to improve the match between the model and the flight test data. The models were done for the same flight conditions as in the third paper. These flight test Mach numbers varying from 0.85 to 1.30 and altitudes varying from 5000 ft. to 25 000 ft. Very good results are obtained with fits between the estimated and the measured signals and correlation coefficients higher than 90%.

Submitted to the following journal:

1-De Jesus Mota, S. Nadeau Beaulieu M., Botez R. M., 2007, *Structural deflections on a F/A-18 aircraft structure following flight flutter tests expressed with a MIMO State Space model constructed using the subspace identification method,* AIAA Journal of Aircraft, March 2007

To be presented to the following conferences:

1-De Jesus Mota, S. Nadeau Beaulieu M., Botez R. M., 2007, *Structural deflections on a F/A-18 aircraft structure following flight flutter tests expressed with a MIMO State Space model constructed using the subspace identification method,* AIAA Modeling and simulation Conference and Exhibit, Hilton Head, South Carolina, Aug 20-23 2007

CHAPTER 10

STRUCTURAL DEFLECTIONS ON A F/A-18 AIRCRAFT STRUCTURE FOLLOWING FLIGHT FLUTTER TESTS EXPRESSED WITH A MIMO STATE-SPACE MODEL CONSTRUCTED USING THE SUBSPACE IDENTIFICATION METHOD

S. De Jesus Mota [1], M. Nadeau-Beaulieu[2] and R. Botez[3]
Ecole de technologie supérieure, Montréal, Québec, Canada, H3C 1K3

M. Brenner [4]
NASA Dryden Flight Research Center, Edwards, CA

10.1 Abstract

The aim of this paper is to determine the mathematical relationship (model) between control deflections and structural deflections of the F/A-18 modified aircraft. Five sets of signals from flight flutter tests corresponding to the excited sources were measured by NASA DFRC (Dryden Flight Research Center). These excitations are: differential ailerons, collective ailerons, collective stabilizers, differential stabilizers, and rudders. The signals to be used by the model are of 2 types: control deflections time histories and their corresponding structural deflections. We choose to use the subspace identification method in order to identify the MIMO (Multi Input, Multi Output) system. Nonlinear inputs are used to fit the outputs signals. We apply this method for a number of sixteen flight conditions for which the Mach number varies from 0.85 to 1.30 and the altitudes vary from 5,000 ft to 25,000 ft. Very good results are obtained with a fit between the

---

[1]Research assistant, sandrine.de-jesus-mota.1@ens.etsmtl.ca.

[2]PhD student, michel.nadeau-beaulieu.1@ens.etsmtl.ca, AIAA Member.

[3]Professor, ruxandra@gpa.etsmtl.ca, AIAA Member.

[4]Aerospace Engineer, Martin.J.Brenner@nasa.gov.

estimated and the measured signals above 73% and a correlation coefficient higher than 90% for most cases.

10.2 Introduction

The work presented in this paper uses flight flutter test data from NASA DFRC (Dryden Flight Research Center). For more details on these data, the reader is invited to consult Lind et al. (1999) and Voracek et al. (2003). In this paper, a model is presented which computes the structural deflection of the flexible F/A-18 following a given control input. The model is build using the subspace parameter estimation methods from flight flutter tests. The subspace identification algorithm, a very efficient non-iterative system identification technique, is used for the mathematical model identification and is explained extensively by Ljung (1999). The next sections present a literature review on system identification methods, and in particular, on the subspace method used in aeroelasticity applications.

The Autoregressive Moving Average Method (ARMA) and the Neural Networks Theory were used by Sung et al. (2005) to identify the flutter behaviour of a transonic wing. Kukreja and Brenner (2006) later used the Non-Linear Autoregressive Moving Average EXogenous (NAMAX) model to study the flutter dynamics of a pitch-plunge system subjected to limit cycle oscillations. The dynamics of a flexible wing model has also been modelled by Silva and Vartio (2006) by use of the impulse response method and the Eigensystem Realisation Algorithm (ERA). An output-error minimisation method was performed based on a large flexible aircraft by LeGarrec et al. (2001).

The subspace method was already applied in many other fields such as fibre optics research by Galvao et al. (2005). In the aerospace field, the subspace method has been used to identify the effects of the aircraft's control surfaces motion on the rigid modes of the F/A-18 from flight flutter tests by Brenner et al. (1997). In this method, the ailerons were excited by use of Schroeder frequency sweeps. The accelerations of the aircraft

were filtered using a wavelet transform and the aircraft's roll responses were both identified in time and frequency domains.

In the present study, the subspace identification method is used to identify the structural deflections of the F/A-18 aircraft's surfaces from flight flutter tests.

10.3 Methodology

The methodology part of this paper has three distinct sections:

10.2.1: Description of the Schroeder excitation inputs signals and data pre-processing
10.2.2: Description of the architecture of the state-space models
10.2.3: Extended description of the subspace system identification method.

10.3.1 Flight Flutter test data preprocessing

In order to obtain the recorded flight flutter tests data, the flight control computer (FCC) for the F/A-18 aircraft was modified by adding a Research Flight Control System (RFCS) to generate the Schroeder frequency sweep control inputs. The software used by the RFCS to control the actuators was called the On Board Excitation System (OBES). The Schroeder frequency sweep generated by the OBES is a large number of harmonics, equally spaced in the frequency domain. An example of the OBES control inputs time history is shown in figure 10.1.

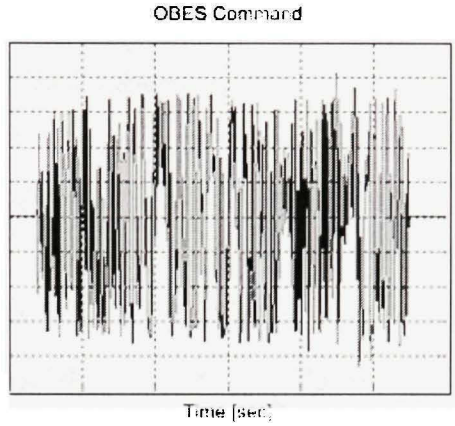

Figure 10.1 ***OBES control inputs versus time***

The OBES Schroeder excitation signal is defined in equation (10.1):

$$OBES(t) = \sum_{k=1}^{C} A_k \sin\left(2\pi f_k t + \phi_k\right) \tag{10.1}$$

where f_k is the k^{th} measurement frequency, ϕ_k is the k^{th} phase and A_k is the k^{th} amplitude of the Schroeder signal. Details on the theory of Schroeder signals are given by Schroeder (1970). The OBES generated Schroeder signal is sent to the aircraft actuators to generate the F/A-18 control surfaces oscillations. For each flight test record, the excited control surfaces may be one of the following: *differential aileron*, *collective aileron*, *collective stabilizer*, *differential stabilizer* or *rudders*. The outputs of the mathematical model are the structural deflections of both wings and trailing edge flaps. The tests were performed for a combination of Mach numbers from 0.85 to 1.30 and for altitudes from 5,000 ft to 25,000 ft. At each flight condition, characterized by an altitude and a Mach number, all the five different manoeuvres above mentioned were performed to generate different records with a time length of 30 seconds. In order to capture all the system dynamics when building the mathematical model, each manoeuvre corresponding to a given altitude and Mach number was concatenated to generate a single long time record.

In this paper, we use firstly the measured accelerations provided by NASA DFRC laboratories. The measured accelerations on the structural surfaces are very noisy. We remove the noise in order to identify the F/A-18 model by performing a double integration on the surface accelerations to obtain the surface deflections. Since the noise is a random process with a mean zero value, then the integration removes any noise contribution to the data. Therefore, no additional filter is required. The filtering effect of the accelerations double integration is shown in figure 10.2, where speeds and positions time histories are represented for the left wing surface.

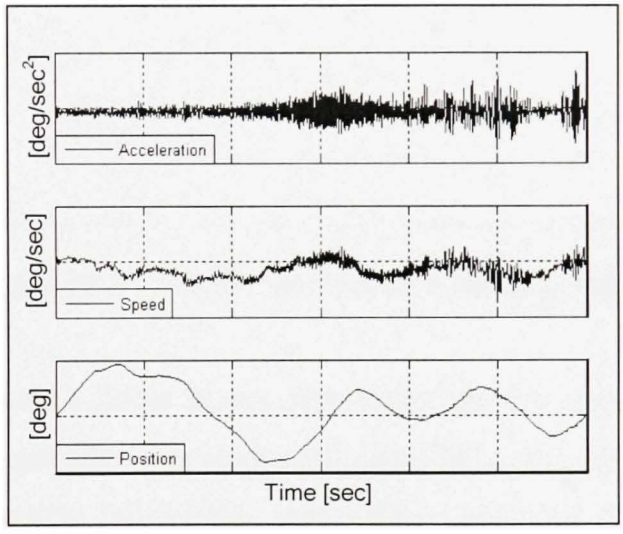

Figure 10.2 ***Left wing accelerations and their single and double integrations with time, which give the speeds and the deflections with time***

In figure 10.2, only the structural surface deflection acceleration for the *left wing* is shown to illustrate the way in which integrations operations remove the unwanted noise.

10.3.2 State-space model architecture description

The linear MIMO (Multiple Input Multiple Output) model representing the system is presented in figure 10.3.

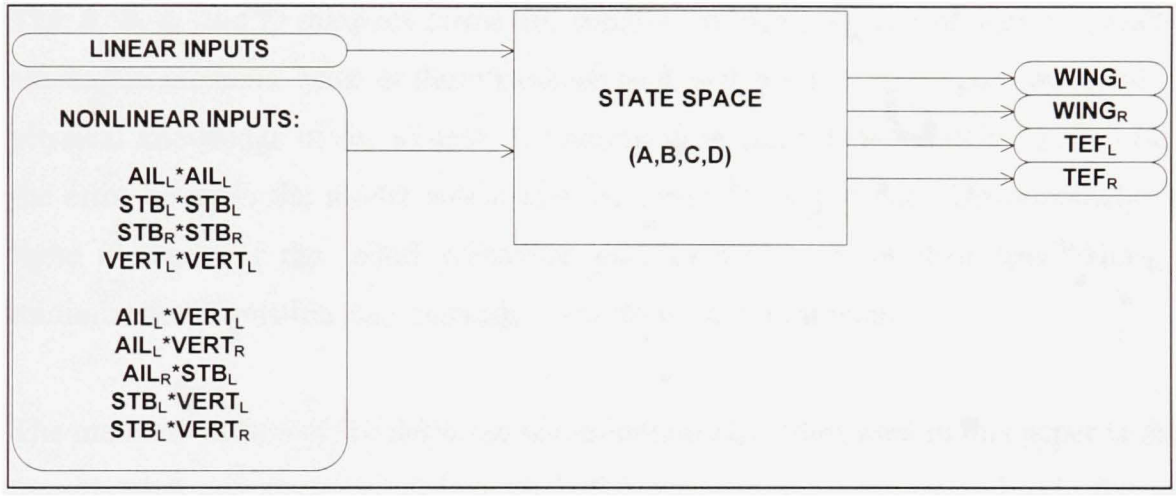

Figure 10.3 ***MIMO model with nonlinear inputs***

In figure 10.3, the model's linear inputs, represented by the upper left block of figure 10.3, are the left and right aileron positions AIL_L and AIL_R, the stabilizer positions STB_L and STB_R and the vertical tail $VERT_L$ and $VERT_R$ positions. These inputs are also combined with non-linear inputs of second degree in order to improve the match between the model and the data. The outputs are the wings deflections $WING_L$ and $WING_R$ and the trailing edge flaps TEF_L and TEF_R deflections. The state-space matrices *A, B, C* and *D* are identified with the subspace method which will be explained in the following section.

10.3.3 Description of the subspace system identification algorithm

Generally, a discrete linear model is defined with the following equations (10.2) and (10.3):

$$x(t+\Delta T)_{n\times 1} = A_{n\times n}x(t)_{n\times 1} + B_{n\times m}u(t)_{m\times 1} + w(t)_{n\times 1} \tag{10.2}$$

$$y(t)_{o\times 1} = C_{o\times n}x(t)_{n\times 1} + D_{o\times m}u(t)_{m\times 1} + v(t)_{o\times 1} \tag{10.3}$$

The A, B, C and D matrices terms are usually estimated by use of various parameter estimation methods. Most of these methods start with a set of initial guesses found from physical knowledge of the system. A minimization algorithm is further used to reduce the error between the model output and the given flight test data. Unfortunately, with these methods, if the initial parameter guesses are far from their true values, the minimization algorithm may converge towards a local minimum.

The main advantage of the subspace identification algorithm used in this paper is that is non-iterative and does not require an initial guess of the parameters inside the state-space matrices [A, B, C, D]. The only information required by the subspace method is the input and the output data vectors.

The subspace identification algorithm is implemented in the Matlab System Identification Toolbox. The basic theory of this algorithm is described by Ljung (1999) and the manner in which the algorithm is implemented in Matlab® is given by Ljung (2006). The main concept of the subspace method is the definition of the system observability matrix Γ_r in the following equation (10.4) from modern control theory, where r represents a forward prediction horizon. This matrix can be obtained from the system inputs $u(t)$ and outputs $y(t)$ and its expression is the following:

$$\Gamma_r \overset{def}{=} \begin{bmatrix} C_{o\times n} \\ CA_{o\times n} \\ \cdots \\ CA^{r-1}{}_{o\times n} \end{bmatrix}_{ro\times n} \tag{10.4}$$

Once this observability matrix Γ_r is known, the state space matrices $[A,\ B,\ C,\ D]$ are obtained by use of the input and output vectors. The detailed procedure to obtain the observability matrix and the discrete state space matrices will further be explained. The theory regarding the subspace algorithm is divided into five sections. The first section (10.2.3.1) defines basic description of matrices and equations necessary to its demonstration, sections (10.2.3.2) and (10.2.3.3) explain the two steps necessary to calculate the observability matrix given in equation (10.4) while the fourth section (10.2.3.4) explains the obtention of the discrete state space matrices $[A,B,C,D]$ from the observability matrix. Finally, the fifth section (10.2.3.5) discuss the model order selection.

10.3.3.1 Basic description of input and output matrices

In order to understand the subspace algorithm, is necessary to define a number of important matrices. The input data obtained later than the reference time t can be arranged into a *Hankel* matrix as follows:

$$U_f \overset{def}{=} \begin{bmatrix} u[t]_{m\,x\,1} & u[t+\Delta t]_{m\,x\,1} & u[t+2\Delta t]_{m\,x\,1} & \dots & u[t+(j-1)\Delta t]_{m\,x\,1} \\ u[t+\Delta t]_{m\,x\,1} & u[t+2\Delta t]_{m\,x\,1} & u[t+3\Delta t]_{m\,x\,1} & \dots & u[t+j\Delta t]_{m\,x\,1} \\ \dots & \dots & \dots & \dots & \dots \\ u[t+(r-1)\Delta t]_{m\,x\,1} & u[t+r\Delta t]_{m\,x\,1} & u[t+(r+1)\Delta t]_{m\,x\,1} & \dots & u[t+(r+j-2)\Delta t]_{m\,x\,1} \end{bmatrix}_{mn\,x\,N} \tag{10.5}$$

In this matrix, the subscript f stands for *future* inputs due to the fact that only the inputs given later than at time t are included into the matrix. The subscript N represents the length of the measured output vector and the index j is a dummy variable adjusted such that all data available in the identification are included on each line of the U_f matrix. Please note that if the expression in the brackets of the matrix element u has a value greater than N, the value of U_f is zero. Similar *Hankel* matrices can be constructed with

the output vectors and defined as Y_f. The future state vector is defined by the use of the following matrix:

$$Y_f = \left[x[t]_{n\,x\,1} \quad x[t+\Delta t]_{n\,x\,1} \quad \ldots \quad x[t+(j-1)\Delta t]_{n\,x\,1} \right]_{n\,x\,N} \tag{10.6}$$

Is also necessary to define the extended following controllability matrix:

$$\Delta_r = \left[\left(A^{r-1}B\right)_{n\,x\,1} \quad \left(A^{r-2}B\right)_{n\,x\,1} \quad \ldots \quad (AB)_{n\,x\,1} \quad B_{n\,x\,1} \right]_{n\,x\,r} \tag{10.7}$$

and finally the impulse response matrix is given by:

$$H_r = \begin{bmatrix} D_{o\,x\,m} & 0 & 0 & 0 & 0 \\ CB_{o\,x\,m} & D_{o\,x\,m} & 0 & \ldots & 0 \\ CAB_{o\,x\,m} & CB_{o\,x\,m} & D_{o\,x\,m} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ CA^{r-2}B_{o\,x\,m} & CA^{r-3}B_{o\,x\,m} & CA^{r-4}B_{o\,x\,m} & \ldots & D_{o\,x\,m} \end{bmatrix}_{ro\,x\,Nm} \tag{10.8}$$

It is also important to describe the noise effect on the *future* output of the system. This noise effect is defined by use of the *Hankel* matrix V:

$$V \overset{def}{=} \begin{bmatrix} V_1[t]_{o\,x\,1} & V_1[t+\Delta t]_{o\,x\,1} & V_1[t+2\Delta t]_{o\,x\,1} & \ldots & V_1[t+(j-1)\Delta t]_{o\,x\,1} \\ V_2[t+\Delta t]_{o\,x\,1} & V_2[t+2\Delta t]_{o\,x\,1} & V_2[t+3\Delta t]_{o\,x\,1} & \ldots & V_2[t+j\Delta t]_{o\,x\,1} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ V_k[t+(r-1)\Delta t]_{o\,x\,1} & V_k[t+r\Delta t]_{o\,x\,1} & V_k[t+(r+1)\Delta t]_{o\,x\,1} & \ldots & V_k[t+(r+j-2)\Delta t]_{o\,x\,1} \end{bmatrix}_{ro\,x\,N} \tag{10.9}$$

where the value of the term V_k is defined with equation (10.10):

$$V_k = CA^{k-2}w(t) + CA^{k-3}w(t+1) + \ldots + Cw(t+k-2) + v(t+k-1) \tag{10.10}$$

In the subspace algorithm, the *future* value of the outputs is related to the future value of the states and input by use of the following equations:

$$X_f = A^r X_p + \Delta_r U_p \tag{10.11.1}$$

$$Y_f = \Gamma_r X_f + H_r U_f + V \tag{10.11.2}$$

Equations (10.11.1) and (10.11.2) were then derived from a state-space equations (10.2) and (10.3). The next section will show how the observability term $\Gamma_r X_f$ can be isolated from equation (10.11.2).

10.3.3.2 Removal of input and noise contribution to the outputs

The first step to obtain the observability matrix Γ_r is to isolate the term $\Gamma_r X_f$ in equation (10.11.2). This procedure can be divided into two steps: 1. An orthogonal projection done to remove the input contribution $H_r U_f$ and 2. An instrument variable approach to remove the noise contribution V. The algorithm used to perform this procedure is called *instrumental variables approach* and is thoroughly explained by Ljung (1999) and Viberg et al. (1995, 1997). This part of the algorithm is described in following sections A) and B).

A) Orthogonal projection to remove the input contribution

In order to remove the inputs contributions $H_r U_f$, a geometric interpretation of equation (10.11.2) is used as shown in figure 10.4 and also explained by Galvao et al.(2005).

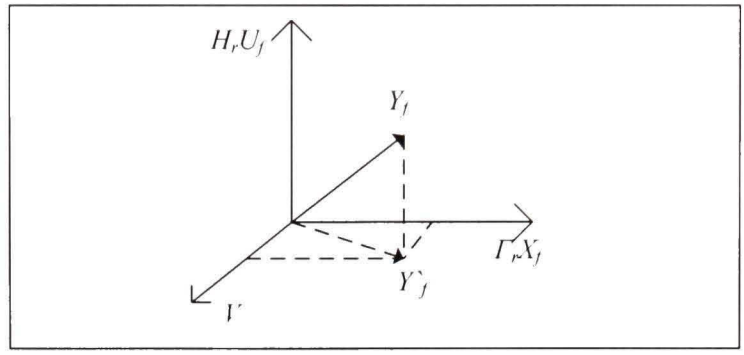

Figure 10.4 : ***Perpendicular projection of the future outputs perpendicular to the future inputs***

If one interprets equation (10.11.2) as a vector, the output contribution can be removed by projecting the output vector Y_f perpendicular to the input contribution $H_r U_f$, which may be achieved with the following projection matrix:

$$\left[\Pi_{U^T}^{\perp}\right]_{N \times N} = I - U_f^T \left(U_f U_f^T\right)^{-1} U_f \tag{10.12}$$

A detailed proof concerning this orthogonal projection matrix is available from Galvao paper (2005). If this perpendicular projection operator is applied on U_f, it is equivalent to find the projection of $H_r U_f$ perpendicular to U_f which, logically, is zero. Mathematically, it can be shown by use of equation (10.13):

$$U_f \Pi_{U^T}^{\perp} = U_f - U_f U_f^T \left(U_f U_f^T\right)^{-1} U_f = U_f - U_f I = 0 \tag{10.13}$$

Post-multiplying both sides of equation (10.11a) by the projection operator $\Pi_{U^T}^{\perp}$ yields:

$$\left[Y_f \Pi_{U^T}^{\perp}\right]_{r \times N} = \Gamma_r X_f \Pi_{U^T}^{\perp} + H_r U_f \Pi_{U^T}^{\perp} + V \Pi_{U^T}^{\perp} = \left[\Gamma_r X_f \Pi_{U^T}^{\perp}\right]_{r \times N} + \left[V \Pi_{U^T}^{\perp}\right]_{r \times N} \tag{10.14}$$

The input contribution has now been removed from the prediction equation.

B) Instrumental variable to remove the noise contribution

It is now necessary to exclude the noise term $V\Pi_{U^T}^{\perp}$ of equation (10.14). This can be done by post-multiplying equation (10.14) with a suitable matrix Φ that is not correlated to the noise matrix V. Let's define the matrix Φ as follows:

$$\Phi = [\varphi_s(t) \quad \varphi_s(t+1) \quad \ldots \quad \varphi_s(t+j-1)]_{s \times N} \tag{10.15}$$

where φ_s are vectors that are uncorrelated with the noise. The number of lines s of this matrix may have any value, but it has to be higher than the desired order n of the dynamic system. This variable is called an *instrument variable* as defined by Ljung (1999) and Viberg (1997) and it is used to reduce the noise term of equation (10.14) to zero. Let's post-multiply equation (10.14) by the transpose of Φ and normalize with the number of sample (by dividing the equation by N) in the data as follows:

$$[G]_{ro \times s} \overset{def}{=} \frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T + \frac{1}{N} V \Pi_{U^T}^{\perp} \Phi^T \overset{def}{=} [\Gamma_r T_N]_{ro \times s} + [F_N]_{ro \times s} \tag{10.16}$$

In equation (10.16), the subscript N implies that the value of T_N and F_N are approximated for a data record containing N data points. In order to cancel out the noise term without affecting the term dependant on the future states T_N, the requirement for a proper *instrument* matrix Φ is that it must be correlated to the future states X_f, but uncorrelated with the noise term, mathematically, it can be expressed by the following equations:

$$\lim_{N \to \infty} F_N = \lim_{N \to \infty} \frac{1}{N} V \Pi_{U^T}^{\perp} \Phi^T = 0 \tag{10.17}$$

$$\lim_{N\to\infty} T_N = \lim_{N\to\infty} \frac{1}{N} X_f \Pi^{\perp}_{U^T} \Phi^T = T \tag{10.18}$$

In equation (10.18), the parameter T is equivalent to the estimation of T_N with an infinite number of data points. Equation (10.17) implies that as the number of samples N goes to infinity, the noise matrix and the *instrument matrix* Φ must cancel each other and F_N goes to zero. The equation (10.16) can therefore be summarized as:

$$G = \frac{1}{N} Y_f \Pi^{\perp}_{U^T} \Phi^T \approx \Gamma_r T \tag{10.19}$$

The remaining step of the above demonstration is to find an appropriate *instrument* matrix Φ. As it was stated before, the first requirement for the *instrument* matrix Φ is that it must be correlated with the future states, which are unknown at this point. Even though these states are unknown, their value is dependant on the past inputs and outputs, these past inputs and outputs are therefore a possible choice. The second requirement is that the matrix Φ must be uncorrelated with the noise. This is always the case when the system inputs are properly selected. This is because the noise can be seen as the error between the model and the data and, for a good model, this error is completely random. The past input and outputs are therefore a logical choice for the *instrument* matrix. In this paper, the *instrument* matrix used is described in equation (10.20) which was taken from Ljung (1999).

$$\Phi = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}_{s \times N} = \begin{bmatrix} y[t-1]_{o\times 1} & y[t]_{o\times 1} & \cdots & y[t+j-2]_{o\times 1} \\ \cdots & \cdots & \cdots & \cdots \\ y[t-h]_{o\times 1} & y[t-h+1]_{o\times 1} & \cdots & y[t-h+j-1]_{o\times 1} \\ - & - & - & - \\ u[t-1]_{m\times 1} & u[t]_{m\times 1} & \cdots & u[t+j-2]_{m\times 1} \\ \cdots & \cdots & \cdots & \cdots \\ u[t-h]_{m\times 1} & u[t-h+1]_{m\times 1} & \cdots & u[t-h+j-1]_{m\times 1} \end{bmatrix}_{s \times N} \tag{10.20}$$

In this equation, the subscript p stands for *past* input and outputs and the parameter h is the number of past inputs and outputs used by the algorithm. Once a proper instrument matrix has been found, the output equation can be described with equation (10.19). Inserting equation (10.16) into equation (10.19) yields:

$$\frac{1}{N} Y_f \Pi_{U^T}^{\perp} \Phi^T = \frac{1}{N} \Gamma_r X_f \Pi_{U^T}^{\perp} \Phi^T \tag{10.21}$$

At this point, every term of the left hand side of equation (10.21) are known. The next step will be to extract the observability matrix Γ_r. This can be done by performing a Singular Value Decomposition (*SDV*). More details on this procedure will be explained in section 10.2.3.3.

10.3.3.3 Determination of the Observability matrix from Singular Value Decomposition

Once the matrix G is known from equation (10.19), it is possible to decompose it into three sub-matrices using a well known linear algebra theorem called *Singular Value Decomposition*. This theorem is explained in through details in reference by Patel et al. (1993) and is expressed as follows:

$$G_{\text{ro x s}} = U_{\text{ro x ro}} S_{\text{ro x s}} V_{\text{s x s}}^T \tag{10.22}$$

In equation (10.22), the matrix S is made of the singular values of G. These singular values are the positive square root of the eigenvalues of $\left(G^T G\right)$, these eigenvalues are sorted in descending order from the first row to the last row of matrix S. The matrices U and V are called singular vectors. They are respectively the orthonormal eigenvectors of $\left(GG^T\right)$ and of $\left(G^T G\right)$. The following demonstration will show how the new matrices

defined by the *Singular Value Decomposition* can be used to obtain an extended observability matrix Γ_r that relates properly the inputs to the outputs. The singular value decomposition provides a possible combination of matrices whose product gives the matrix G. Of these two matrices, the first one is a possible observability matrix and the second one is a possible state vector. The singular value decomposition of equation (10.22) applies for a dynamic system of any order. If the desired order n of the dynamic system is known, a proper procedure separate the first n singular values and singular vectors of the system from the other singular values and vectors. In practice, when performing singular value decomposition, only the system significant singular value and their corresponding singular vectors must be kept. The number of singular values considered corresponds to the assumed order of the model. The singular value that is kept should represent the true dynamic of the system and the small singular value that are not taken into account should correspond to errors due to noise. Once the singular value decomposition is done, it is possible to obtain a proper estimate of the observability matrix Γ_r. Notice that many different combinations of observability matrix and T matrix can lead to a set of parameters value that insures a proper match. According to reference Ljung (1999), it follows that the value of the observability matrix may be expressed with the following equation:

$$\Gamma_r = U \tag{10.23}$$

It is also possible to add weight functions to the matrix G obtained from equation (10.23) before performing singular value decomposition as follows:

$$G_{Weight} = W_1 G W_2 \tag{10.24}$$

The reason for adding weight matrices is to remove any residual error due to noise in the variable G. These weight matrices are made of parameters that are uncorrelated with the projected noise matrix F_N. In the absence of noise, adding a weight matrix has no effect

on the identification results, but it improves the results when the identification is done on noisy data.

After a weight matrix is selected, the new observability matrix is found by use of the following equation:

$$\Gamma_r = W_1^{-1} U \tag{10.25}$$

Many authors have proposed expressions for different weight matrices and they have an influence on the results of the identification. A good summary on the work of different authors on these weight matrices formulations is available from Viberg et al. (1997) where different expressions for weight matrices are derived using the same mathematical approach. For original work on the different weight matrices, the reader is invited to consult the papers written by Verhaegen (1994), Van Overschee (1994), Viberg (1995) and Larimore (1990). In this paper, the weight matrices defined by Larimore (1990) were tried and gave proper results. Since there is not much noise in the data related to this project, the algorithm is not very sensitive to the selected weight and it wasn't necessary to try other weight formulations to obtain good results. The weights as defined by Larimore are defined by use of the following equation:

$$W_1 = \left(\frac{1}{N} Y \Pi_{U^T}^{\perp} Y^T \right)^{-1/2} ; W_2 = \left(\frac{1}{N} \Phi \Pi_{U^T}^{\perp} \Phi^T \right)^{-1/2} \tag{10.26}$$

At this point the observability matrix Γ_r has been determined from equation (10.25). This observability matrix can now be used to obtain the value of the matrices [*A*, *B*, *C*, *D*]. The procedure to do it will be shown in the next section.

10.3.3.4 Determination of the system matrices [A, B, C, D] from the observability matrix

A) Estimating *A* and *C*

Once the observability matrix is known, it is quite easy to obtain the estimates of the A and C matrices. If we refer to equation (10.4) as it was defined at the beginning of the theory section, the estimate matrix C is obtained by taking the following terms of the observability matrix:

$$\hat{C} = \Gamma_{(1:o,1:n)} \tag{10.27}$$

Equation (10.27) simply means that we have to extract the first o lines and n columns of the observability matrix. The matrix A can be found from the observability matrix by solving the following equation:

$$\Gamma_{(o+1:ro,1:n)} = \Gamma_{(1:o(r-1),1:n)} A \tag{10.28}$$

In this equation, the left hand side represents the observability matrix with the first sub-matrix $\hat{C}$ removed and the right hand side represents the observability matrix with the last sub-matrix CA^{r-1} removed. This is equivalent to the following equation:

$$\begin{bmatrix} CA_{o\times n} \\ CA^2{}_{o\times n} \\ \cdots \\ CA^{r-1}{}_{o\times n} \end{bmatrix}_{(r-1)o\times n} = \begin{bmatrix} C_{o\times n} \\ CA_{o\times n} \\ \cdots \\ CA^{r-2}{}_{o\times n} \end{bmatrix}_{(r-1)o\times n} A_{n\times n} \tag{10.29}$$

In this equation, the only unknown is the state matrix A. If the modified observability matrices $\Gamma_{(o+1:ro,1:n)}$ and $\Gamma_{(1:o(r-1),1:n)}$ of equation (10.29) were both square matrices, it would be easy to find $\hat{A}_{n\times n}$ by just pre-multiplying both side of the equation by the inverse of

$\Gamma_{(1:o(r-1),1:n)}$. Since these matrices are not necessarily square, equation (10.29) can be solved by use of the following equation:

$$\hat{A}_{n\times n} = \left(\begin{bmatrix} C_{o\times n} \\ CA_{o\times n} \\ \cdots \\ CA^{r-2}{}_{o\times n} \end{bmatrix}^{\dagger} \right)_{n\times(r-1)o} \begin{bmatrix} CA_{o\times n} \\ CA^{2}{}_{o\times n} \\ \cdots \\ CA^{r-1}{}_{o\times n} \end{bmatrix}_{(r-1)o\times n} \tag{10.30}$$

In equation (10.30), the superscript "$\dagger$" denotes the Moore-Penrose pseudo-inverse described by Viberg (1995). This is a more general type of inversion which does not require the matrices to be square. The pseudo-inverse can be computed by singular value decomposition and reader is invited to consult references from Patel (1993) and Klema (1980) for more details.

B) Estimating *B* and *D*

Once the system matrices $\hat{A}$ and $\hat{C}$ are known, it is possible to estimate the B and D matrices by use of a linear regression technique. The discrete state-space model of equation (10.2) and (10.3) can be converted into a discrete transfer function by using the discrete operator z defined with the following equation:

$$\begin{aligned} zx(t) &= x(t+\Delta t) \\ z^{-1}x(t) &= x(t-\Delta t) \end{aligned} \tag{10.31}$$

Which leads to the following transfer function:

$$\hat{y}(t \mid B, D) = \hat{C}\left(zI - \hat{A}\right)^{-1} Bu(t) + Du(t) \tag{10.32}$$

Where $\hat{y}(t \mid B, D)$ means the *estimated* output which depends on the values of the matrices B and D. A very efficient way to find the unknown parameters B and D of equation (10.32) is to use a linear regression method. The estimated output $\hat{y}(t)$ may be expressed by the following equation:

$$\hat{y}(t)_{o \times 1} = \eta(t)\hat{\theta} = \eta(t)_{o \times (nm+om)} \begin{bmatrix} Vec(B) \\ Vec(D) \end{bmatrix}_{(nm+om) \times 1} \tag{10.33}$$

In equation (10.33), the matrix $\eta(t)$ is made of the past and presents inputs $u(t)$. The single column vector $\hat{\theta}$ represents all the estimated parameters to be found by the regression, these unknown parameters are all the elements of the matrices B and D. The operator "Vec" builds a column vector from a matrix by stacking its columns on top of each other.

10.3.3.5 Selection of the mathematical model order

The subspace method explained above applies for a state-space model of any order. Recall that the order of a State-Space model is defined by the rank of matix A defined in equation 10.2 and this order must be carefully selected in order to obtain an appropriate model. The order should be high enough to insure that the model represents all the important dynamics of the system, however, choosing an order that is too high may lead to an overfitting problem. Overfitting occurs when the match between the model's outputs and the flight flutter test data is very good for data used by the identification algorithm, but is poor for fresh data set aside for the validation process. A good approach for model order selection is therefore to insure that for a given order, the model performance is optimal for both the identification test data and another data set aside for the validation process. In this paper, there was no validation flight test data to evaluate the model's generality. To solve this problem, robustness test data were generated by slightly perturbing the initial identification flight test data using a re-

sampling technique. These robustness test data were used as validation data. The method used to generate these robustness test data is explained in details in section 10.4.1.3 below. Different model orders were tried and for each order, the model performance on the identification flight test data and the robustness test data were evaluated. It was found that the optimal model order was eleven (11).

10.4 Results

This section explains the way in which the non-linear state-space mathematical models obtained from the subspace system identification method are evaluated. The accuracy of the mathematical model is demonstrated with four different criterions:

10.4.1 The criterions used to evaluate the model

10.4.2 The match between the model output and the output from the flight test data shown graphically for one flight condition.

10.4.3 The summary of results obtained for all flight conditions.

10.4.4 The graphical representation of the worse results

The results section is divided into four parts: the first part explains the criterion used to evaluate the model. These criterions are the correlation and the fit coefficients, and the robustness test. The second part shows graphically the match between the model outputs and the input data for a given flight condition. The third part summarizes the results for all flight conditions by use of the average correlation coefficients and fit coefficients. The last part shows graphically the worse obtained results.

10.4.1 Criterion used to evaluate the model

10.4.1.1 Correlation coefficient

The first method used to validate the model is the correlation coefficient. The correlation coefficient R can be interpreted as a measure of the scatter in the graph shown in Figure 10.510.5 between the measured output and the calculated (estimated) output.

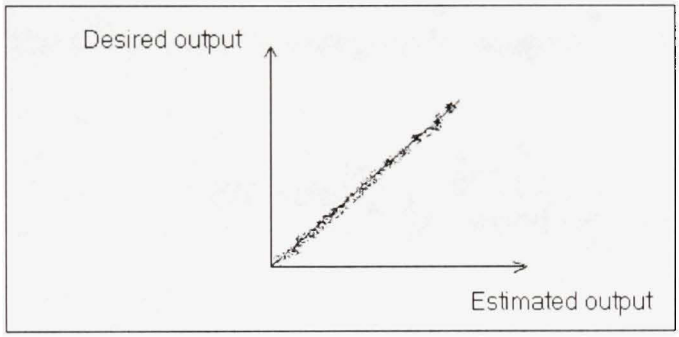

Figure 10.5 ***Visual interpretation of the measured versus the estimated structural surface deflection outputs***

Mathematically, R (which has values between -1 and 1) is given by the following equation:

$$R = \frac{Cov(y,\hat{y})}{\sqrt{Var(y)Var(\hat{y})}} \tag{10.34}$$

where Cov is the covariance, Var is the variance, y is the measured output and $\hat{y}$ is the estimated output. The correlation coefficient R equal to one (R = 1) denotes *perfect linear dependency* (no scatter) between the measured and the calculated or estimated outputs. A correlation coefficient equal to minus one (R = -1) denotes *inverse linear dependency* between the measured and the estimated outputs. A correlation coefficient of zero (R = 0) denotes the *linear independency* between the measured and the estimated

outputs. The correlation coefficient computes the goodness of the model in a statistical sense, but provides little information about the model error. More information can be obtained by the second method: the *fit* coefficient.

10.4.1.2 Fit coefficient

The fit coefficient is defined as 100 % multiplied by the ratio between the L_2-norm of the error between the data and the model over the L_2-norm of the error between the data and its mean value. The *fit* coefficient is expressed by equation (10.35):

$$FIT = 100\left(1 - \frac{\|y - \hat{y}\|}{\|y - mean(y)\|}\right) \tag{10.35}$$

Where the term in $(\|y - \hat{y}\|)$ is the L_2-norm of the error. The L_2-norm is defined with equation (10.36):

$$\|y - \hat{y}\| = \frac{1}{2}\sqrt{\sum_{1}^{n}(y - \hat{y})^2} \tag{10.36}$$

Equation (10.35) therefore becomes:

$$FIT = 100\left(1 - \sqrt{\frac{\sum_{1}^{n}(y - \hat{y})^2}{\sum_{1}^{n}(y - mean(y))^2}}\right) \tag{10.37}$$

In equation (10.37), the term under the square root represents the ratio between two residuals: the residual between the model and the data and the residual between the data

and their mean value. Intuitively, the fit coefficient represents the data variation percentage explained by the model. The main advantage of using the *fit* coefficient is that it takes into account the data variation about its mean in order to evaluate the model quality. For example, even if a model has an output very close to the data output, it will have a poor *fit* coefficient if there are much more small oscillations in the data than in the model.

10.4.1.3 Robustness test

Normally, the best manner to test the performance of a model would be its use in a simulation with a set of input data which was not used in the parameter identification process. The resulting model output is then compared to the flight test data outputs corresponding. This comparison can be done by using the correlation and fit coefficients and this test is called cross-validation or *acid test*. Unfortunately, in this paper, it was not possible to set data aside for the validation because there was only one set of data records available for each given input excitation, altitude or Mach number. For this reason, we decided to evaluate the robustness of our estimated model by considering the model's output resulting from a simulation with slightly perturbed input signals. The purpose of this test was to evaluate the effect on the model's output of negligible input signals perturbations. If the model is not sensitive to very small perturbation of its inputs, it indicates that the model is robust and would more likely become an *acid test*. The difference between a perturbed input signal and the initial input signal is illustrated in figure 10.6.

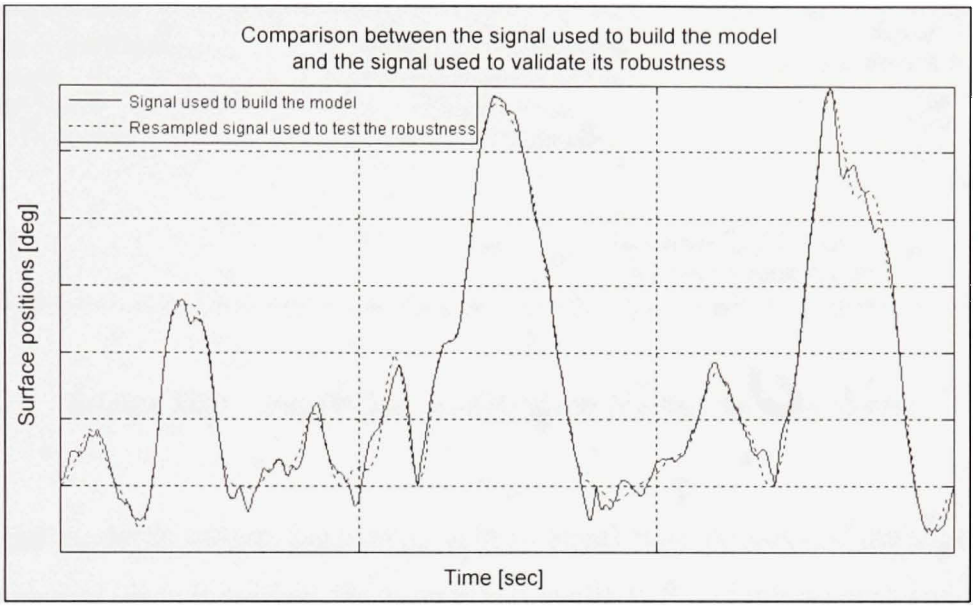

Figure 10.6 ***Perturbed signal used for the robustness test***

The perturbed input signal shown in figure 10.6 was generated by performing the following operations:

-Resample the signal by keeping only one point for a given number b of points.

-Reconstruct the signal from these points by performing interpolations in order to obtain the initial sampling rate.

This procedure is illustrated in figure 10.7 where Δt is the sampling rate.

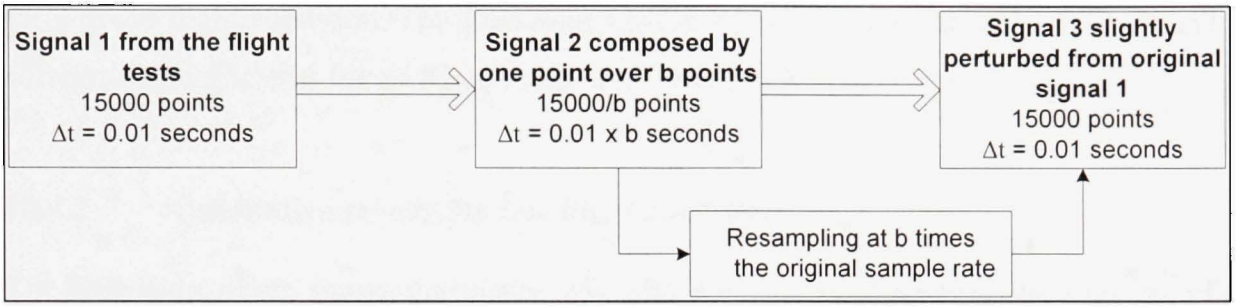

Figure 10.7 ***Input data modification for the robustness test***

Thus, if the model is robust, must react well to small perturbations of the input signals. This means that there is neither divergence nor oscillations on the output signals and the output fit parameters must be similar to the fit of the signal used in the identification.

The robustness test was used to select the most appropriate non-linear inputs to be used and to build the models. These modified data sets were also used to select the most appropriate model order. Recall from figure 10.3 that the model's inputs are the different control surfaces deflections and linear combinations of these deflections. The use of a high number of non-linear inputs may cause the model to *overfit* the data. When this situation occurs, there is a small model error compared to the data set used to build it, but the model performs poorly on these new data sets.

Since we do not have a separate data set to use in the cross-validation, we decided to use the modified input data for the robustness test above described as added non-linear input would likely cause an over fitting problem. We chose to add only the non-linear inputs which improved the model performance on both the original data and the modified data used in the robustness test. In each case, the model performance was calculated in term of correlation and fit coefficients.

The following section will demonstrate qualitatively the model performance by showing the time history of the model output with respect to the time history of the flight test data

for a given flight condition. The following section shows the mean results of the fit and correlation coefficients for all flight conditions considered in this paper.

10.4.2 Qualitative results for one flight condition

The following figure shows the results obtained for the model identification by use of the subspace method for both sets of data used in the identification and in the robustness test. These results are for a Mach number of 0.85 and an altitude of 5,000 ft. In figure 10.8, the full line represents the measured output from the flight test data and the stars represent the model outputs.

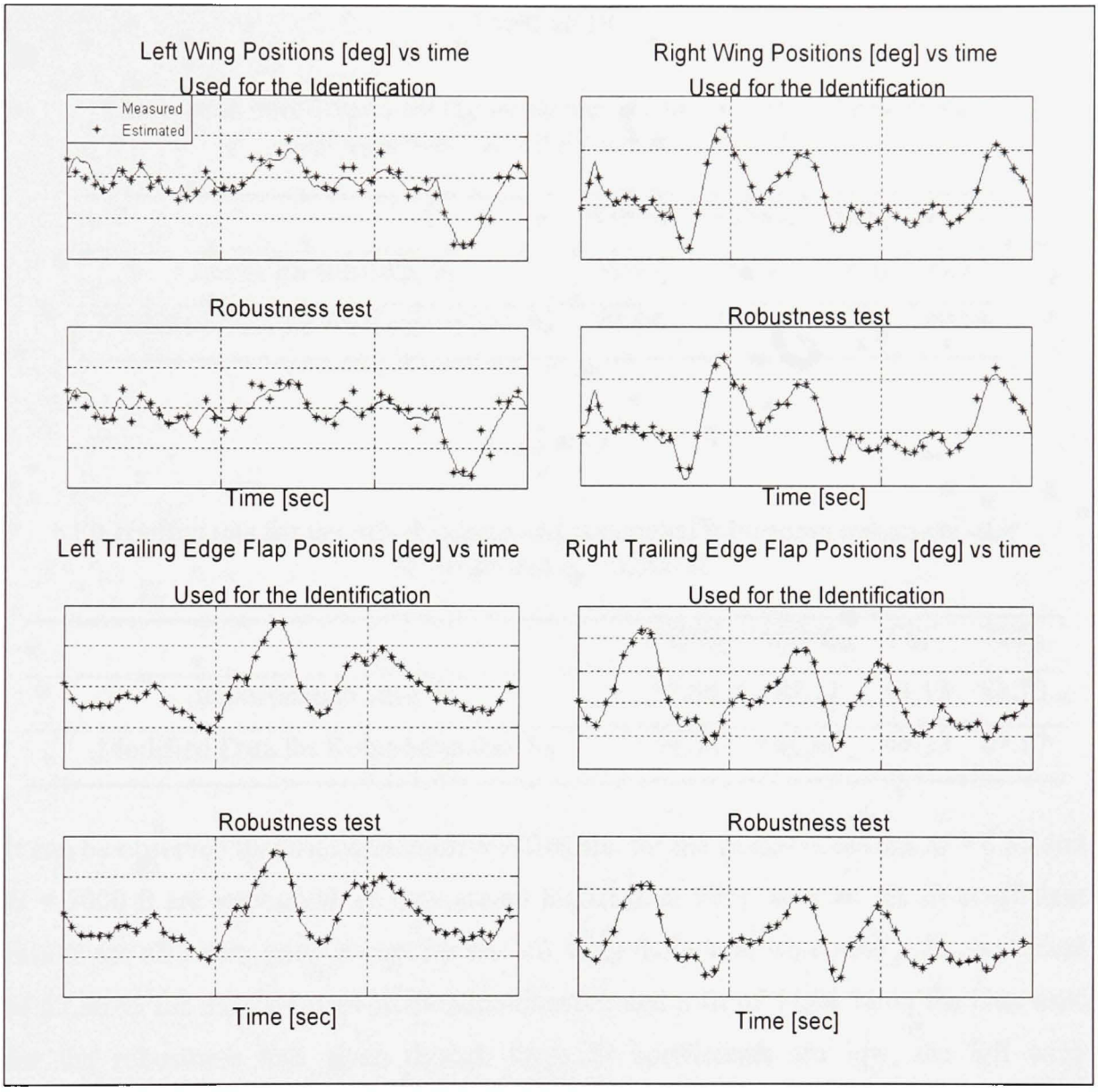

Figure 10.8 ***MIMO model identification and robustness test for M = 0.85 and H = 5,000 ft***

From a visual inspection, is clear that each model outputs match very well the flight test data for both the original data and the data modified for the robustness test for the aircraft different surfaces. Table 10.1 shows the correlation coefficients between the estimated and the measured outputs.

Table 10.10

Correlation coefficients for the initial signals and re-sampled robustness test signals for $M = 0.85$ and $H = 5{,}000$ ft

	$WING_L$	$WING_R$	TEF_L	TEF_R
Identification data, %	92.09	99.28	99.61	99.37
Modified Data for Robustness test %	90.88	98.23	99.26	99.18

Table 10.2

Fit coefficients for the initial signals and resampled robustness test signals for $M = 0.85$ and $H = 5{,}000$ ft

	$WING_L$	$WING_R$	TEF_L	TEF_R
Identification data, %	57.56	87.11	91.13	88.73
Modified Data for Robustness test %	54.04	80.66	87.33	87.17

It can be observed that the correlation coefficients, for the flight conditions $M = 0.85$ and $H = 5000$ ft are very good, as they are all higher than 90%. Most of the fit coefficient results are also very good except for the left wing deflection which has a fit coefficient of 57.56 % for the data used in the identification and a fit of 54.04 % on the data used for the robustness test. Even though these fit coefficients are low, the left wing deflection time history graphs for estimated and measured results are still close, as seen on figure 10.8. Another observation which can be made is that there is not a high difference between the model accuracy on the data used in the identification with respect to the modified data for the robustness test.

10.4.3 Results summary for all flight conditions

The results above shown in Tables 10.1 and 10.2 are given for a single flight condition characterized by a Mach number of 0.85 and an altitude of 5000 ft. In this paper, F/A-18

aircraft model with different set of parameters were identified for a number of 16 different combinations of Mach numbers and altitudes, as seen on Table 10.3.

Table 10.3

Fits and Correlation coefficients for all 16 flight test conditions

Flight Test Conditions	Outputs	Identification		Validation	
		Correlation	Fit	Correlation	Fit
$M = 0.85,\quad H = 5000\text{ft}$	$WING_L$	92.09	**57.56**	90.88	**54.04**
	$WING_R$	99.28	87.91	98.23	80.66
	TEF_L	99.61	91.13	99.26	87.33
	TEF_R	99.37	88.73	99.18	87.17
$M = 0.85,\quad H = 10000\text{ft}$	$WING_L$	97.45	76.41	97.23	75.43
	$WING_R$	99.68	91.51	99.20	86.23
	TEF_L	99.53	90.28	98.68	84.42
	TEF_R	99.77	93.20	99.33	88.67
$M = 0.85,\quad H = 15000\text{ft}$	$WING_L$	98.64	80.56	97.68	77.98
	$WING_R$	97.20	75.12	96.59	66.89
	TEF_L	99.62	91.23	99.09	86.47
	TEF_R	99.33	88.39	99.07	86.51
$M = 0.90,\quad H = 5000\text{ft}$	$WING_L$	95.39	**68.36**	95.25	**61.15**
	$WING_R$	99.60	91.04	99.55	90.17
	TEF_L	99.88	94.15	99.61	87.95
	TEF_R	99.37	88.79	99.09	86.63
$M = 0.90,\quad H = 10000\text{ft}$	$WING_L$	96.97	73.63	97.41	75.62
	$WING_R$	98.42	81.87	97.16	75.34
	TEF_L	99.65	91.56	99.11	86.55
	TEF_R	99.46	89.04	98.72	82.33
$M = 0.90,\quad H = 15000\text{ft}$	$WING_L$	91.54	**54.12**	77.45	**30.36**
	$WING_R$	99.28	87.83	95.15	68.99
	TEF_L	99.69	92.06	97.76	77.35
	TEF_R	99.84	94.27	95.89	71.58
$M = 1.10,\quad H = 10000\text{ft}$	$WING_L$	99.23	87.21	98.85	84.20
	$WING_R$	99.43	89.34	99.16	87.19
	TEF_L	99.70	92.21	99.17	86.41
	TEF_R	99.54	90.32	99.38	88.56

Table 10.3 (suite)

Fits and Correlation coefficients for all 16 flight test conditions

$M = 1.10, \quad H = 15000\text{ft}$	$WING_L$	98.29	81.04	97.81	72.94
	$WING_R$	99.88	94.94	99.69	88.92
	TEF_L	99.58	90.90	98.90	86.26
	TEF_R	99.90	95.46	99.60	86.91
$M = 1.10, \quad H = 20000\text{ft}$	$WING_L$	99.27	87.89	99.12	86.33
	$WING_R$	99.55	90.49	99.33	87.96
	TEF_L	99.66	91.80	99.40	89.15
	TEF_R	99.75	92.95	99.68	91.57
$M = 1.10, \quad H = 25000\text{ft}$	$WING_L$	99.13	86.00	99.01	84.88
	$WING_R$	99.80	93.60	99.69	91.75
	TEF_L	99.30	88.24	98.89	84.81
	TEF_R	99.80	93.57	99.47	88.90
$M = 1.20, \quad H = 10000\text{ft}$	$WING_L$	99.82	93.92	99.60	91.11
	$WING_R$	98.90	85.09	98.91	85.15
	TEF_L	99.84	94.16	99.46	89.30
	TEF_R	99.41	89.12	98.63	83.13
$M = 1.20, \quad H = 15000\text{ft}$	$WING_L$	98.44	80.77	97.13	73.98
	$WING_R$	99.76	92.95	98.62	82.70
	TEF_L	99.70	92.04	97.68	76.95
	TEF_R	98.51	82.14	97.70	77.75
$M = 1.20, \quad H = 20000\text{ft}$	$WING_L$	99.53	89.81	99.12	85.99
	$WING_R$	99.87	94.58	99.81	93.30
	TEF_L	99.37	88.71	99.07	86.27
	TEF_R	99.77	93.16	99.36	88.15
$M = 1.20, \quad H = 25000\text{ft}$	$WING_L$	99.11	86.12	99.21	87.16
	$WING_R$	99.74	92.80	99.66	91.69
	TEF_L	99.65	91.57	99.49	89.70
	TEF_R	99.81	93.80	99.55	90.31
$M = 1.30, \quad H = 20000\text{ft}$	$WING_L$	98.80	84.35	98.51	82.23
	$WING_R$	98.62	82.54	98.41	78.19
	TEF_L	99.69	92.09	99.46	89.77
	TEF_R	98.65	83.49	97.70	78.29
$M = 1.30, \quad H = 25000\text{ft}$	$WING_L$	98.33	77.49	97.99	74.95
	$WING_R$	99.35	88.55	99.14	85.62
	TEF_L	98.24	77.94	97.68	74.85
	TEF_R	99.78	93.38	98.39	81.14

The results in **bold** in table 10.3 are the worse results. The following observations can be made from these results:

1-The correlation coefficient between the model output and the flight test data output is always near 100 % and re-sampling the data for the robustness test has a very small effect on the correlation coefficient.

2-The fit coefficient is higher than 80 % for every structural surface except the left wing which is slightly lower and modifying the data for the robustness test also has a small effect.

These good results for the correlation and the fit coefficients indicate that the model is accurate. There is a very small degradation of the fit and correlation coefficients when the inputs re-sampled inputs for the robustness test are used, which indicates that the model is robust and does not over-fit the data. Please note that the model is slightly less accurate in the left wing deflection prediction.

10.4.4 Worse results

From the table 10.3 above, we notice that three flight conditions do not well satisfy the previous conclusions and they are: $M = 0.85$, $H = 5000$ ft; $M = 0.90$, $H = 5000$ ft and $M = 0.90$, $H = 15000$ ft. The estimated and measured outputs related to the worse flight conditions cases are shown in figure 10.9.

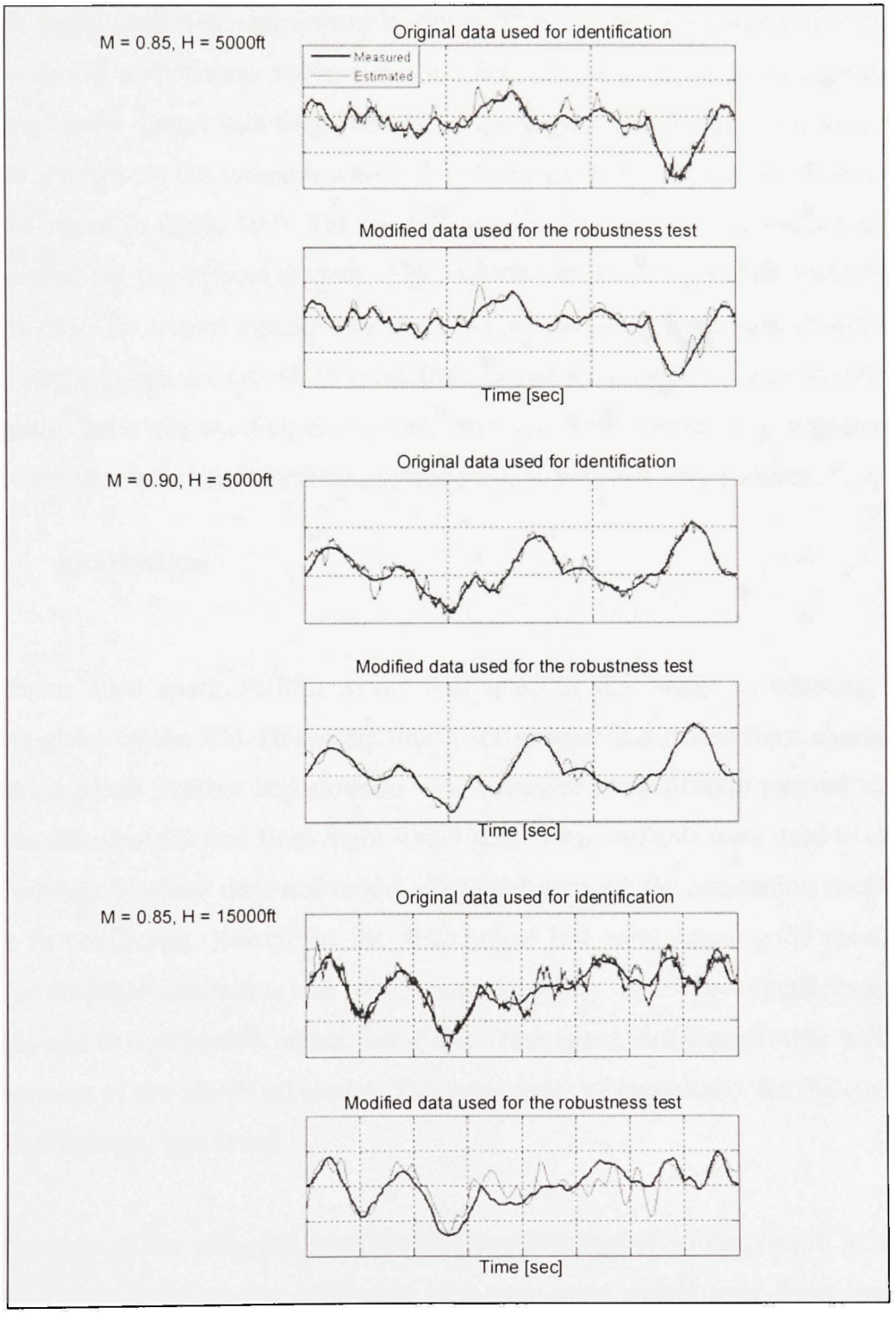

Figure 10.9 : ***The three worse cases for the left wing deflections outputs***

For each flight condition represented in figure 10.9, the left wing deflection from the model is shown with respect to the deflection from the flight test data. In each case, the deflection model output was found by use of the inputs from the original data and the modified input from the robustness test. We observe that there exist oscillations of the identified signal in figure 10.9. Yet, we can say that the trends of the measured signals are respected by the estimated ones. The oscillations do not generate instabilities or divergences of the output signals. Therefore, we noted that the worse results affect only the first output of the system which is the left wing position ($WING_L$). For the robustness test signals, there are oscillations too but they are less marked. The robustness test signals have the same trends as the measured signals, which is very positive.

10.5 Conclusions

A nonlinear state space MIMO model was used in this study to estimate surface positions given by the F/A-18 control inputs for sixteen flight conditions characterized by different Mach number and altitudes. The subspace identification method was used for the model identification from flight flutter tests. Two methods were used to calculate the differences between data and model-identified outputs: the correlation coefficients, and the fit coefficient. Except for the three worse left wing cases, good results were found for all flight conditions and were characterized by correlation coefficients higher than 96% and fit coefficients higher than 73%. Then, resampled signals were used to test the robustness of the identified model. The same order of magnitude, for the correlation and fit coefficients, was found.

The advantage of the subspace identification method applied to our model is its small computing time and also the estimation of a very good model only from flight tests inputs and outputs. The estimated model was found to be robust by application of the re-sampling technique. We conclude from the obtained results, that the subspace approach method is very convenient for model identification from flight flutter tests.

10.6 Acknowledgments

Financial support for the work related to this paper was given by the Natural Sciences and Engineering Research Council of Canada NSERC and by the Ministère du Développement économique, de l'Innovation et de l'Exportation MDEII.

10.7 References

Brenner Martin, Eric Feron, April 1997, "Wavelet analysis of F/A-18 aeroelastic and aeroservoelastic flight test data", 38th AIAA/ASME/ASCE/AHS/ASC Structure, Structural Dynamics and Material Conference Exhibit, Kissimmee, Florida, 7-10 April, 1997. , pp. 691-713

Galvao, Roberto K. H., Sillas Hadjiloucas, Victor M. Becerra and John W Bowen, "Subspace system identification framework for the analysis of multimoded propagation of THz-transient signals", Institute of Physics Publishing, Measurement Science and Technology, Vol. 16, pp. 1037-1053, 2005.

Klema Virginia C., Alan J. Laub, 1980, "The Singular Value Decomposition: Its Computation and Some Applications", IEEE Transaction on Automatic Control, vol. AC-25, no. 2, pp. 164-176

Kukreja, Sunil L., Martin Brenner M., "Nonlinear Aeroelastic system identification with application to experimental data", AIAA Journal of Guidance, Control and Dynamics, Vol. 29, No. 2, 2006.

Larimore, Wallace E., 1990, "Canonical variate Analysis in Identification, Filtering, and Adaptive Control" , In proc. 29th IEEE Conference on Decision and Control, Honolulu, Hawaii, December 1990

Le Garrec C., Humbert M., Bucharles A., Vacher P., "In flight aeroelastic model identification and tuning of a flight control system on a large civil aircraft", CEAS International Forum on Aeroelasticity and Structural Dynamics, Madrid, Spain, 5-7 June, 2001.

Lind, Rick, Marty Brenner, (1999), *Robust aeroservoelastic stability analysis: Flight Test Applications*, Advances in industrial control, Springer-Verlag Ed, 204 pp.

Ljung. Lennart , 1999, *System Identification Theory for the user*, Prentice Hall, Upper Saddle River, N.J., 2nd edition, pp.340-351

Ljung Lennart, 2006, *System Identification Toolbox for use with Matlab®*, User Guide Version 6, The Mathworks Inc., 416 pp.

Patel, RajniV., Alan J. Laub, Paul M. Van Dooren., 1993, *Numerical Linear Algebra Techniques for System and Control*, IEEE Press, City College of New York, 724 pp

Silva Walter A., Eric Vartio, Anthony Shimko, Raymond G. Kvaternik, Kenneth W. Eure and Robert C. Scott, Development of Aeroservoelastic Analytical Models and Gust Load Alleviation Control Laws of a SensorCraft Wind-Tunnel Model Using Measured Data, 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Newport, USA, 4-6 May, 2006.

Sung Won Kok, Her Mann Tsai, Mani Sadeghi, Feng Liu, "Non-linear impulse methods for aeroelastic simulations", 23rd AIAA Applied Aerodynamics Conference, Toronto, Canada, 6-9 June 2005, pp. 1-19.

Van Overschee Peter, Bart De Moor, 1994, "N4SID: Subspace algorithm for the identification of combined deterministic-stochastic systems", Automatica, 30(1), pp. 75-93

Verhaegen, Michael , 1994, "Identification of the deterministic part of MIMO state space models, given in innovation form from input-output data", Automatica, 30(1), pp. 61-74

Viberg, Matt, 1995, "Subspace-based Methods for the identification of Linear Time-invariant Systems", Automatica, 31(12), pp. 1835-1852

Viberg, Matt., Bo Wahlberg, and Bjorn Ottersen, 1997, "Analysis of state space system identification methods based on instrumental variables and subspace fitting", Automatica 33(9) pp. 1603-1616

Voracek D., Pendleton E., Reichenback E., Griffin K., Welch L., 2003, *The active aeroelastic wing phase I Flight Research through January 2003*, NASA Dryden Flight Research Center, NASA/TM-2003-210741, 25 pp.

CONCLUSION

In this thesis, two distinct class of methods were used to model the behaviour of a flight vehicle: the *Grey Box* semi analytical method and the *Black Box* subspace system identification method. The main contribution of this thesis consisted in the application of these approaches in solving different problems related to flight vehicles system identification. This section first summarises the results found using these two different approaches before exposing a more general conclusion on the choice of a method depending on the problems characteristics.

11.1 *Grey Box* semi-analytical method

The semi analytical model was applied in Chapter 4 of this thesis to develop a ground dynamics model for the B-427 helicopter. In this model, a spring defined with stiffness and damping coefficients was used to calculate the normal forces on the helicopter at touchdown. Friction equations were used to model the speed decay of the helicopter following touchdown while the friction coefficient was a function of the helicopter oscillations and its speed. The rolling moments (L) and pitching moments (M) were calculated by multiplying a lever arm to the force in z-direction until the skids were completely on the ground, a torsional spring was further used to model the oscillations of the helicopter in roll and pitch at the end of the landing record. The yawing moment at touchdown was computed with a damping term which varied with the normal force between the ground and the helicopter. This ground dynamics model was validated with landing flight test data for one engine inoperative landings and autorotation landings. The model was successfully implemented in a level D global flight model and the model's outputs lie within the FAA tolerance bands for a helicopter simulator qualification.

11.2 *Black Box* subspace system identification method

Different state-space models whose parameters were identified by use of the subspace system identification method were used to model the flight vehicles behaviour in two different applications:

For the first application the helicopter B-427 a state-space model with non-linear inputs was used to model the main rotor torque, tail rotor torque, engine torque and main rotor speed with time. During each flight test record, the pilot applied 2311 multi-step control inputs to excite all modes of the helicopter's motion. The resulting outputs along with the pilot's control inputs were recorded in time, and these records were used to build and validate the model.

There were four types of pilot control inputs: collective, longitudinal cyclic, lateral cyclic and pedal. The model was identified with the subspace method and refined with a Levenberg-Marquardt minimization algorithm. This model was implemented as a simulation and as a prediction.

In the simulation implementation, the model inputs, measured from flight test data, were the control inputs from the pilot and the helicopter's linear and angular velocities. In the prediction implementation, the same type of inputs were used along with the currently measured value of the output in order to predict the future value of the output within a given prediction horizon.

In the simulation implementation, the model's outputs were found to be within the FAA tolerance bands more than 97 % for each output when the subspace method was refined with the Levenberg-Marquardt minimization algorithm. In the prediction implementation, the model error was even lower, and decreased when the prediction horizon was shortened.

On the second application, concerning the F/A-18 Active Aeroelastic Wing research aircraft (Chapter 8 and 10), the deflections of different structural surfaces of the aircraft were determined following an harmonic control input applied on the aircraft control surfaces by the Flight Control Computer. These harmonic inputs may be represented mathematically by Schroeder frequency sweeps.

In the first paper on the F/A-18 aircraft (Chapter 8), nine third order linear state space models MISO were used to estimate the structural surfaces deflections given by the F/A-18 differential ailerons control inputs. The deflections of each structural surface were represented by a separate MISO model.

The inputs of each model were the ailerons control inputs and the deflections of the other structural surfaces. The model was identified with the subspace system identification method and validated for 19 different combinations Mach numbers and altitudes and gave very good values of correlation coefficients and fit coefficients. The estimated model was also found to be robust by the re-sampling technique application.

The second paper concerning the F/A-18 (Chapter 10) represents an improvement with respect to the third one. In this paper, a single Multiple Inputs Single Outputs (MIMO) model was used to estimate simultaneously the structural deflections of the four following surfaces: the left wing, the right wing, the left training edge flap and the right trailing edge flap. The model was built using five different Schroeder excitations: differential ailerons, collective ailerons, collective stabilizers, differential stabilizers, and rudders. As in the previous paper, the mathematical model was represented by a state-space model, but, this time, non-linear inputs were added to the linear inputs to improve the match between the model and the flight test data. The models were conceived for the same flight conditions as the one given in the third paper with Mach numbers that varies from 0.85 to 1.30 and altitudes varying from 5000 ft to 25 000 ft. Very good results were

obtained with a fit between the estimated and the measured signals and correlation coefficients higher than 90%.

11.3 General conclusions

In this thesis, it was shown that a semi-analytical *grey box* approach or a *black box* approach could both be used in the successful identification of mathematical models representing flight vehicle behaviour. This research also confirmed the advantages and drawbacks of each method. In the ground dynamics model, this research presented the following aspects:

- There was an a-priori knowledge of the helicopter behaviour at touchdown. The penalty method used to model the helicopter skids as a spring with a stiffness and damping coefficients had already been used in the literature by Johnson (1997). All other forces and moments on the helicopter were also derived based on the system physics.
- There were only very few landing cases available for the identification.
- A general model was ensured to give reasonable results for other landing conditions not covered by the available landing data.

For all these reasons, a *grey box* semi-analytical model was more suitable than a *black box* model in order to model the ground dynamics of the B-427 helicopter at touchdown. Recall from the introduction that an analytical model, contrary to a *black box* model, was predictable outside the flight conditions used in the training, and was required in the ground dynamics model.

In the second paper, where it was required to estimate the B-427 main rotor torque, tail rotor torque, engine torque and main rotor speed, the problem presented the following aspects:

- There was not enough knowledge available on the physical behaviour of this very complex system.
- There were many time history data available for the identification and validation (409 records for the identification and 138 for the validation).

In this case, it was more appropriate to use a *black box* model because there was a lack of information about the system's physics, and many flight test records at different flight conditions were available to provide a general model.

In the third and fourth papers, it was required to model the aeroelastic structural deflections of the F/A-18 research aircraft. In this case, there was also very little knowledge on the aeroelastic behaviour of this specific aircraft as no structural information necessary to calculate its aeroelastic behaviour analytically was available. It was therefore necessary to use a *black box* system identification method. Unfortunately, there was only one time history data available and it was not possible to perform an *acid test* to validate the model. An alternative validation was done by re-sampling the data and performing a robustness test (see Chapters 8 and 10). From all these observations, it could be concluded that both the semi-analytical method and the *Black box* subspace identification methods were very efficient methods which could be used to solve different problems related to flight vehicle identification.

Therefore, the most important contributions in this thesis were underlined in the presented papers as follows: in the first paper, a new ground dynamics helicopter model was analytically developed and validated with flight tests data, in the second paper, the subspace method was used for the first time for main rotor torque prediction for a helicopter by use of flight tests data and in the third and fourth paper, the subspace method was used for the first time for modeling aircraft flexible deflections from flight flutter tests data. The main challenge in this thesis was the aircraft and helicopter model identification by use of real flight test data and flight flutter test data.

RECOMMANDATIONS

The research presented in this thesis could be improved in many ways or be used as a starting point to undertake many other research projects. This section will list some possible future research directions based on this thesis:

12.1 Ground Dynamics Model

1- The ground dynamics model implemented in this paper was already certified with respect to the FAA tolerance bands requirements and implemented in a global level D flight simulator model. The next step of the simulator certification would be the installation of the mathematical model in a real flight simulator hardware and ask a pilot to land the helicopter simulator using the ground dynamics model developed in this thesis. The pilot would further qualitatively assess the accuracy of the model in reproducing his feelings in the cockpit in a real landing.

2- The ground dynamics model could be further generalized by use of new landing cases which were not available in this research. These cases could include landing on wet runway, grass, slopes, etc.

3- The same model structure could be modified and validated on different helicopters such as helicopters with welled landing gears.

12.2 Model for the simulation and prediction of the main rotor torque, tail rotor torque, engine torque and main rotor speed (Chapter 6)

1-This model was built based on forward flight records with 2311 multi-step inputs excitations. It could be further generalized by use of records representing other flight conditions and pilot's manoeuvres.

2-The model could be identified using other *black box* methods such as Neural Networks or Fuzzy Logic, and results obtained by these methods could be compared.

3-The model could be used to design an envelope protection flight control system in order to limit the maximum and minimum values of the model outputs.

12.3 Models for the identification of the F/A-18 structural surface deflections

1-It would be suitable to perform a higher number of flight flutter tests with the aim of model generalization.

2-Another *black box* system identification method, such as Neural Networks, should be used and its performance should be compared to the performance of the subspace method application.

3-The identified model could be used to design a controller to minimize the structural surfaces deflections.

REFERENCES

Akaike, Hirotsugu., 1969, *Fitting autoregressive models for prediction,* Ann. Inst. Stat. Math., Vol. 21, pp. 243-347

Akaike, Hirotsugu, 1974, "A new look at the statistical model identification", IEEE trans. Automatic Control, AC-19, pp. 716-723

Baraff, David, July 1989, "Analytical method for dynamic simulation of non-penetrating rigid bodies", Association for Computing Machinery's Special Interest Group on Graphics and Interactive Techniques, Volume 23, Issue 2, 1989, pp.223-232

Baraff, David, "Non-penetrating rigid body simulation", Association for Computing Machinery's Special Interest Group on Graphics and Interactive Techniques, 1989, pp.223-232.

Baraff, David, July 1993, "Non-penetrating rigid body simulation", Eurographics Technical report series, 24 pp.

Beer, Ferdinand P.P., E. Russel Johnston Jr., and Elliot R. Eisenberg, George H. Staab, 1999, *Vector Mechanics for Engineers*, Static and Dynamics, McGraw-Hill Science/Engineering/Math, 7th Edition, McGraw-Hill Science/Engineering/Math, 624 pages.

Brenner Martin, Eric Feron, April 1997, "Wavelet analysis of F/A-18 aeroelastic and aeroservoelastic flight test data", 38th AIAA/ASME/ASCE/AHS/ASC Structure, Structural Dynamics and Material Conference Exhibit, Kissimmee, Florida, 7-10 April, 1997. , pp. 691-713

Crisan, Emil G, 2005 , *Validation of a mathematical model for the Bell 427 helicopter using parameter estimation techniques and flight test data*, Master Thesis, Montréal, École de Technologie Supérieure, 124 p.

Demuth, Howard, Mark Beale, Martin Hagan, 2006, *Neural Network Toolbox for use in MATLAB®*, User Guide Version 5, The Mathworks, 848 p.

Federal Aviation Administration, 1994, *Helicopter Simulator Qualification*, Advisory Circular AC 120-63, Federal Aviation Administration, US department of transportation

Galvao, Roberto K. H., Sillas Hadjiloucas, Victor M. Becerra and John W Bowen, "Subspace system identification framework for the analysis of multimoded propagation of THz-transient signals", Institute of Physics Publishing, Measurement Science and Technology, Vol. 16, pp. 1037-1053, 2005.

Guendelman, Eran, Eric Bridson, Ronald Fedkiw, 2003, "Non-convex rigid bodies with stacking", Association for Computing Machinery's Special Interest Group on Graphics and Interactive Techniques, Volume 22 issue 3 , pp. 871-878.

Horn, Joseph, May 1999, *Flight Envelope Limit Detection and Avoidance*, PhD Thesis, Atlanta Georgia Institute of Technology, 110 p.

Horn, Joseph, Calise, A.J., Prasad, J.V.R., October 2002, "*Flight Envelope Limit Detection and Avoidance for Rotorcraft*", Journal of American Helicopter Society, Volume 47, Issue 4, pp. 253-262

Horn, Joseph, Nilesh Sahani, 2004 , "Detection and Avoidance of Main Rotor Hub Moment Limits on Rotorcraft", Journal of Aircraft, Vol. 41, No. 2, pages 372-379

Howitt, Jeremy, 1995, "Carefree Handling for super-agility", American Helicopter Society 2nd Aeromechanics Specialist Conference, Fairfield County, Oct. 11-13 1995, pages 4-58

Jategaonkar, Ravindras V., 2006, *Flight vehicle system identification: A time domain Methodology*, AIAA Progress in Aeronautics and Astronautics, Arlington: Frank K. Lu, 534 pp.

Jategaonkar, Ravindras V., 2006, *Flight Vehicle System identification In Time Domain*, AIAA Professional Development Tutorial, Keystone Resort and Conference Center, Keystone, CO

Jeram, Geoffrey J., 2002, "Open design for helicopter active control system", American Helicopter Society 58th Annual Forum, Montreal, Canada, June 11-13 2002.

Jeram, Geoffrey J., Nilesh H. Sahani., J.V.R Prasad, 2005, "Distributing Limit Protection between Autonomous Restraint and Voluntary Tactile Cues ", 43th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 10-13. 2005, pp.1-14

Johnson, Eric N., Paul A. DiBietto, 1997, "Modeling and simulation for small autonomous helicopter development" in AIAA 1997 modeling and simulation technologies conferences, New Orleans, Aug. 11-13, pp.1-11.

Kalman, R.E., 1960, "A new approach to linear filtering and prediction problems", ASME Journal of Basic Engineering, Series 82D, Mar. 1960, pp. 35-45.

Klema Virginia C., Alan J. Laub, 1980, "The Singular Value Decomposition: Its Computation and Some Applications", IEEE Transaction on Automatic Control, vol. AC-25, no. 2, pp. 164-176

Kukreja, Sunil L., Martin Brenner M., "Nonlinear Aeroelastic system identification with application to experimental data", AIAA Journal of Guidance, Control and Dynamics, Vol. 29, No. 2, 2006.

Larimore, Wallace E., 1990, "Canonical variate Analysis in Identification, Filtering, and Adaptive Control", In proc. 29th IEEE Conference on Decision and Control, Honolulu, Hawaii, December 1990

Le Garrec C., Humbert M., Bucharles A., Vacher P., "In flight aeroelastic model identification and tuning of a flight control system on a large civil aircraft", CEAS International Forum on Aeroelasticity and Structural Dynamics, Madrid, Spain, 5-7 June, 2001.

Lind, Rick, Marty Brenner, (1999), *Robust aeroservoelastic stability analysis: Flight Test Applications*, Advances in industrial control, Springer-Verlag Ed, 204 pp.

Ljung. Lennart , 1999, *System Identification Theory for the user*, Prentice Hall, Upper Saddle River, N.J., 2nd edition, pp.340-351

Ljung Lennart, 2006, *System Identification Toolbox for use with Matlab®*, User Guide Version 6, The Mathworks Inc., 416 pp.

Massey, C.P., and Wells P., 1988, "Helicopter carefree handling systems", Royal Aeronautical Society Conference on Helicopter Handling Qualities and Control, London, UK, 15-17 November 1988

McCool, Kelly, Lance A Flitter, and David J. Hass, 1998, "Development and Flight Test Evaluation of a Rotor System Load Monitoring Technology", American Helicopter Society 54th Annual Forum, Washington, 20-22 May 1998, pp. 408-418

Menon P.K., V. R. Iragavarapu, M.S. Walley 1996, "Estimation of Rotorcraft Limit Envelope Using Neural Networks", Annual forum proceedings of American Helicopter Society, Alexandria, Virginia, vol. 2, pp. 1423-1431

Mirtich B., 1996 , "Impulse-based Dynamic Simulation of Rigid Body Systems", Ph.D. thesis, University of California, Berkeley, December, 1996

Moore, M., Wilhelms, J., 1988, "Collision detection and response for computer animation", Computer Graphics (Proc SIGGRAPH), vol.22, pp. 289-298

NASA Dryden Flight Research Center, 2006, "National Aeronautics and Space Administration : Centers : Dryden : Home: News : Research Update : AAW", In NASA webpage online, <http:// www.nasa.gov /centers/ dryden/ news/ ResearchUpdate /AAW /index.html> consulted on January 25th 2007.

Nadeau Beaulieu Michel, Andrei Vladimir Popov, Ruxandra Botez, Adrian Hiliuta, Ruxandra Popescu, Njuki Mureithi, (2005) *Methodology for Aerodynamic Model Generation for the B-427 Helicopter in Hover in ground effect,* Eleventh

Australian International Aerospace Congress, Melbourne, Australia, March 13-17, 2005.

Nelson, Robert C., 1998, *Flight Stability and Automatic Control,* second edition, Mc Graw Hill, pp.102-103.

Patel, RajniV., Alan J. Laub, Paul M. Van Dooren., 1993, *Numerical Linear Algebra Techniques for System and Control*, IEEE Press, City College of New York, 724 pp

Prouty, Raymond W., 1986, *Helicopter Performance Stability and Control,* Malabar, Florida: Krieger Publishing Company, 731 p.

Rugh, Wilson J., 1993 , *Linear system theory,* Prentice Hall Information and System Sciences Series, New Jersey, 356 pp.

Sahani, Nilesh A., December 2005, "*Envelope Protection Systems for Piloted and Unmanned Rotorcraft*", PhD Thesis, University park, Pennsylvania State University, 143 p.

Sahasrabudhe Vineet, Robert Spauling, Alexandre Faynberg, Joseph Horn and Nilesh Sahani, 2002, "Simulation Investigation of a Comprehensive Collective-Axis Tactile Cueing System", in American Helicopter Society 58th annual forum, Montreal June 11-13 2002, pp. 559-568

Sareen, Ashish K., Michael R. Smith, John V. Howard, 1998, "Helicopter skid gear dynamic drop analysis and test correlation", American Helicopter Society 54th Annual Forum, Washington D.C., May 20-22, pp.1267-1274

Schroeder M.R., 1970, "Synthesis of low-peak factor signals and binary sequences with low autocorrelation," *IEEE Trans. Inform. Theory*, Vol. IT-16, Jan. 1970., pp. 85-89

Silva Walter A., Eric Vartio, Anthony Shimko, Raymond G. Kvaternik, Kenneth W. Eure and Robert C. Scott, Development of Aeroservoelastic Analytical Models and Gust Load Alleviation Control Laws of a SensorCraft Wind-Tunnel Model Using Measured Data, 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Newport, USA, 4-6 May, 2006.

Sung Won Kok, Her Mann Tsai, Mani Sadeghi, Feng Liu, "Non-linear impulse methods for aeroelastic simulations", 23rd AIAA Applied Aerodynamics Conference, Toronto, Canada, 6-9 June 2005, pp. 1-19.

Van Overschee Peter, Bart De Moor, 1994, "N4SID: Subspace algorithm for the identification of combined deterministic-stochastic systems", Automatica, 30(1), pp. 75-93

Verhaegen, Michael , 1994, "Identification of the deterministic part of MIMO state space models, given in innovation form from input-output data", Automatica, 30(1), pp. 61-74

Viberg, Matt, 1995, "Subspace-based Methods for the identification of Linear Time-invariant Systems", Automatica, 31(12), pp. 1835-1852

Viberg, Matt., Bo Wahlberg, and Bjorn Ottersen, 1997, "Analysis of state space system identification methods based on instrumental variables and subspace fitting", Automatica 33(9) pp. 1603-1616

Voracek D., Pendleton E., Reichenback E., Griffin K., Welch L., 2003, *The active aeroelastic wing phase I Flight Research through January 2003*, NASA Dryden Flight Research Center, NASA/TM-2003-210741, 25 pp.

Yavrucuk, Ilkay, J. V. R. Prasad and Antony J. Calise, 2001, "Adaptive Limit Detection and Avoidance for Carefree Maneuvering", AIAA Atmospheric Flight Mechanics Conference and Exhibit, Montreal, Aug. 6-9, 2001, pp. 1-9

Yavrucuk, Ilkay., J.V.R. Prasad and Antony J. Calise, 2002, "Carefree Maneuvering Using Adaptive Neural Networks", AIAA Atmospheric Flight Mechanics Conference abd Exhibit, Monterey, CA, Aug. 5-8 2002, pp. 291-299

Yavrucuk, Ilkay, J. V. R. Prasad, 2002, "Adaptive Limit Margin Prediction and Control Cueing for Carefree Maneuvering of VTOL Aircraft", American Helicopter Society Flight Controls and Crew System Design Technical Specialist Meeting, Philadelphia, Oct. 9-11, 2002, pp. 1-7