Advanced Techniques for Flutter Clearance

by

Laurent Guillaume Duchesne

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

January 1997

© Massachusetts Institute of Technology 1997. All rights reserved.

Author .....
Department of Aeronautics and Astronautics
January 17, 1997

Certified by .....
Eric Feron
Assistant Professor
Thesis Supervisor

Accepted by .....
Jaime Peraire
Chair, Graduate Office

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

FEB 10 1997

LIBRARIES

Advanced Techniques for Flutter Clearance

by

Laurent Guillaume Duchesne

Submitted to the Department of Aeronautics and Astronautics
on January 17, 1997, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

In this thesis, a new methodology for flutter boundary prediction using experimental data is developed. Due to the complexity of the aircraft's structure and aerodynamics, it appeared necessary to rely on a simplified wing model to create a low order state space representation that would capture all the essential dynamics of a flutter problem. The methodology is the following: first, a low order finite element model was derived using linear aerodynamic theory and a Pade approximation. The technique also relies on a time-frequency analysis which selectively eliminates noise from the recorded signals in order to estimate the transfer function of the system. A graphical interface was developed to perform this task more efficiently. Then, a state space model parameterized by the dynamic pressure q is identified with a quasi-Newton optimization based on a frequency domain cost function. Finally, the flutter boundary is determined based on the domain of stability of the parameterized model. This methodology has been validated first on a theoretical example, then on wind tunnel data through the Benchmark Active Controls Technology (BACT) model and finally on the F18 System Research Aircraft.

Thesis Supervisor: Eric Feron
Title: Assistant Professor

Acknowledgments

First of all, I would like to thank my advisor Eric Feron for the interest and enthusiasm that he has showed throughout my stay at MIT. I would also like to thank Jim Paduano who acted as a co-advisor for this project. I wish to acknolewdge all the NASA Dryden Flight Research Center engineers with whom I have interacted with a special thank to Marty Brenner with whom I have shared many productive discussions. Finally, I would like to thank all my lab mates who provided useful advice and more importantly friendship.

This work was supported by NASA under the Consortium program NCC 2-5116 entitled "Advanced Techniques for Flutter Clearance" and the cooperative research agreement DFRCU-95-025 entitled "Methods for In-Flight Robustness Evaluation".

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Flutter is a phenomenon which is very critical in aeronautical engineering and arises in the design of the wings and the tail of an aircraft. It involves a coupling between inertial, structural and aerodynamic forces. Under certain conditions, typically in the transonic region or at high dynamic pressure, the combination of these three forces may create a self excited system that becomes unstable. Flutter therefore must be avoided on any aircraft because its presence can result in the yield of the structure. Furthermore, with the rise of new types of structures, wings are becoming more and more flexible thus bringing flutter boundary even closer to normal operating conditions. A more critical point is that, even though flutter boundary can be estimated in the design procedure of an aircraft, the results are not very reliable and a large margin of error must be allowed for. Therefore, flutter clearance is an essential part of the aircraft certification problems which has to be accomplished through extensive and expensive flight tests. The goal of this thesis is to provide efficient and reliable ways to predict flutter boundary in real-time by using recorded flight data.

The wing flutter problem has been heavily studied in the past, and its theoretical development, based on linear aerodynamic theory, can be found in [9] and [37]. A Pade approximation is also used to model some delays in the aerodynamic forces. An algorithm known as the k-method was developed in [36], using strip theory which

improved flutter boundary estimations. Currently, the so-called p-k iteration [13] is used to predict the flutter boundary. This algorithm actually solves finite element equations and gives an estimation of the damping ratio of each mode, given a specific flight condition. Then, an iteration on the flight condition needs to be made to find the flutter boundary. The preceding methods rely essentially on analytical computations and also on assumptions about the accuracy of linear aerodynamic theory. Indeed they do not take into consideration any flight measurements. In the past, experimental data have been used to clear operating points in the flight envelope from flutter but little extrapolation to the flutter boundary was attempted [3]. Lately, some attention has been given to the use of modern control theory such as robustness analysis in the prediction of flutter boundary . In [11], a methodology to obtain a conservative bound of flutter for an airfoil in a wind tunnel is developed. The problem was set up as a real-μ problem with two uncertainties, Mach number and dynamic pressure. The same idea was adapted to the F18-SRA [21] where unmodeled dynamics were incorporated into the uncertainty as well.

1.2 Outline

Classical flutter boundary determination for a typical wing section is described in Chapter 2. The equations of motion are first derived and the flutter boundary is estimated using a damping ratio extrapolation. As further explained, this method may provide poor performance when the recorded data are taken at flight conditions that are not very close to the flutter boundary.

In Chapter 3 a review of major classical system identification techniques is provided since it quickly appears as one of the critical points in a flutter clearance problem. A short description of parametric identification methods is given but more attention is devoted to subspace identification methods. It is also shown how such methods can handle multiple data sets.

The overall procedure proposed in this thesis is developed in Chapter 4. A technique based on time-frequency analysis is described to estimate the transfer function

of the system based on frequency sweep excitation signals. A Newton optimization algorithm is then used to identify the system at different flight points simultaneously. A validation of this method is done using the model described in Chapter 2 since it represents the dynamics of the flutter phenomenon very well.

Application of the identification technique to the BACT model in a wing tunnel experiment was achieved. The flutter boundary determination was duplicated from earlier work by K. Gondoly based on robustness analysis. Improvements of results obtained with experimental data is also presented.

Finally, the proposed procedure is applied to the F18 SRA and described in Chapter 6. The flight data were provided by NASA Dryden Flight Research Center and the experiments included flight conditions at different altitudes (10,000 30,000 and 40,000 feet) and different Mach numbers in the transonic region (Mach 0.8, 0.85, 0.9 and 0.95).

Chapter 2

Classical flutter boundary determination

Although a real aircraft is not strictly speaking a single elastic unit, it is necessary from an engineering view point to treat it as such in order to deal with the complexity of flutter problems. Another simplification is to limit the study to a specific part of the aircraft that is susceptible to generating unstable oscillations. In general, an aircraft has two critical points, the wings and the tail, and it is usually assumed that there is no interaction between those two parts of the aircraft. In this thesis, only the wing flutter problem will be addressed.

It is also necessary to make some further assumptions because, even if the wing is considered as a cantilevered structure, it would still be a continuous system which has an infinite number of modes. Since this is unfeasible in practice, a finite element model is usually derived from the geometry and the material of the aircraft. However, to understand the most important part of the physics involved in a flutter problem, a typical wing section is enough [9]. In this Chapter, only a simple typical section is presented to allow the reader understand the flutter phenomenon.

2.1 Typical section dynamics

Let us consider a unit width strip of a two-dimensional flat plate airfoil which has two degrees of freedom: a bending mode and a torsion mode (also called pitching mode). As a convention, a positive bending h will be downward, and a positive torsion α will be pitching up. The semi-chord of the airfoil is denoted b, and the ratio of the distance between the elastic axis and the center of gravity to the semi-chord is a_h. The x-axis is defined to be parallel to the air speed U. The model and the notations are illustrated in Figure 2-1. Note also that two springs were incorporated in the system to model the strain due to the rest of the wing. For an elementary unit length dx on the x-axis, the small element of the airfoil has an elementary mass denoted dm. The kinetic energy of an element of mass at a distance x from the elastic axis is

$$dT = \frac{1}{2}(\dot{h} + x\dot{\alpha})^2 dm. \tag{2.1}$$

So, the kinetic energy of the typical section is

$$T = \frac{1}{2}(m\dot{h}^2 + 2S_\alpha \dot{h}\dot{\alpha} + I_\alpha \dot{\alpha}^2), \tag{2.2}$$

where m is the mass per unit span of the wing, S_α is the static moment of inertia about the elastic axis and I_α is the mass moment of inertia about the elastic axis. Those terms can be computed using the following expressions:

$$m = \int dm,$$

$$S_\alpha = \int x dm,$$

$$I_\alpha = \int x^2 dm.$$

If the stiffness of the bending and torsion spring are respectively defined as k_h and k_α, the potential energy is

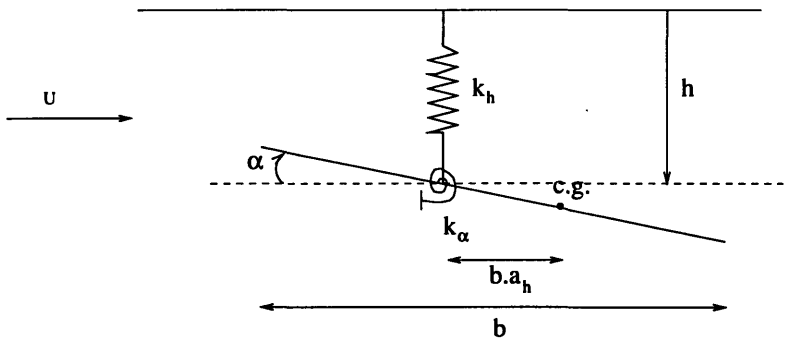


Figure 2-1: Typical section of the airfoil.

$$U = \frac{1}{2}k_h h^2 + \frac{1}{2}k_\alpha \alpha^2. \tag{2.3}$$

Since the gravity does not play a fundamental role in the flutter phenomenon, this force was omitted in the formula for the potential energy for reasons of simplicity. The Lagrange equation of motion can now be computed

$$m\ddot{h} + S\ddot{\alpha} + k_h h = -L \tag{2.4}$$

$$S\ddot{h} + I_\alpha \ddot{\alpha} + k_\alpha \alpha = M, \tag{2.5}$$

where L and M represent respectively the aerodynamic force and moment about the elastic axis.

2.2 Aerodynamic model

Proceeding further in the derivation of the equation of motion, the description of the aerodynamic forces needs to be performed. The theory that will be considered is called linearized aerodynamic theory. It assumes that all the forces and moments are linear with respect to the air density ρ. The forces can be decomposed into two parts: the non-circulatory and the circulatory one.

Defining b as the chord of the wing, a_h as the ratio of the distance from the center

of gravity to the elastic axis and the chord, and U as the airspeed, the lift of the non-circulatory part can be decomposed as follows:

1. A lift force with center of pressure at the mid-chord

$$L_1 = \rho\pi b^2(\ddot{h} - a_h b\ddot{\alpha}) \tag{2.6}$$

2. A lift force with center of pressure at $\frac{3}{4}$-chord point

$$L_2 = \rho\pi b^2 U\dot{\alpha} \tag{2.7}$$

3. A nose down moment

$$M_a = -\frac{\rho\pi b^4}{8}\ddot{\alpha} \tag{2.8}$$

For the circulatory part, the step response to a vertical velocity component w (or downwash) was obtained by Wagner, Kussner, von Karman and Sears and is equal to

$$L_3(\tau) = 2\pi b\rho U w\Phi(\tau), \tag{2.9}$$

where $\tau = Ut/b$ is non-dimensional quantity proportional to the time t. The function Φ is called the Wagner's function. It is a highly nonlinear function but can, however, be approximated by

$$\Phi(\tau) = 1 - 0.165e^{-0.041\tau} - 0.335e^{-0.32\tau}. \tag{2.10}$$

The downwash due to the two degrees of freedom h and α consist of the three following terms:

1. A uniform downwash corresponding to a pitching angle α, $w = U\sin\alpha = U\alpha$

2. A uniform downwash due to vertical translation $\dot{h}$

3. A nonuniform downwash due to $\dot{\alpha}$ whose value is $(\frac{1}{2} - a_h)b\dot{\alpha}$

In the interval $[\tau_0, \tau_0 + t\tau_0]$, the downwash $w(\tau_0)$ increases by an amount $\frac{dw(\tau_0)}{d\tau_0}d\tau_0$. When $d\tau_0$ is sufficiently small, this may be regarded as an impulsive increment and the corresponding circulatory lift is

$$dL_3(\tau) = 2\pi b\rho U\Phi(\tau - \tau_0)\frac{dw(\tau_0)}{t\tau_0}d\tau_0.$$

By the principle of superposition, the circulatory lift becomes

$$L_3 = 2\pi b\rho U\int_{\infty}^{\tau}\Phi(\tau - \tau_0)\frac{w}{\tau_0}(\tau_0)d\tau_0. \tag{2.11}$$

The total lift on the on the typical section is

$$L = L_1 + L_2 + L_3, \tag{2.12}$$

and the total moment about the elastic axis is

$$M = (\frac{1}{2} + a_h)bL_1 + a_h bL_2 - (\frac{1}{2} - a_h)bL_3 + M_a. \tag{2.13}$$

Since the Wagner's function is a function of τ, it is convenient to convert the equation of motion and use τ instead of t as the time variable. To do this, we need to relate the differentiation of a function f with respect to the physical time $\dot{f}$ to the differentiation with respect to the non dimensional time f':

$$\dot{f} = \frac{df}{dt} = \frac{df}{d\tau}\frac{d\tau}{dt} = \frac{U}{b}f'. \tag{2.14}$$

Using this notation, the aerodynamic lift and moment about the elastic axis induced by h and α are

$$\begin{aligned} L(\tau) &= 2\pi b\rho U^2\int_{-\infty}^{\tau}\Phi(\tau - \tau_0)[\alpha'(\tau_0) + \frac{1}{b}h''(\tau_0) + (\frac{1}{2} - a_h)\alpha''(\tau_0)]d\tau_0 \\ &\quad +\rho\pi U^2(h'' - a_h b\alpha'') + \rho\pi bU^2\alpha' \end{aligned} \tag{2.15}$$

and

$$M(\tau) = (\frac{1}{2}+a_h)2\pi b^2\rho U^2\int_{-\infty}^{\tau}\Phi(\tau-\tau_0)[\alpha'(\tau_0)+\frac{1}{b}h''(\tau_0)+(\frac{1}{2}-a_h)\alpha''(\tau_0)]d\tau_0$$
$$+a_h b\rho\pi U^2(h''-a_h b\alpha'')-(\frac{1}{2}-a_h)\rho\pi b^2U^2\alpha'-\frac{\rho\pi b^2U^2}{8}\alpha''. \quad (2.16)$$

Using this notations, the full equation of motion using the dimensionless time is

$$m\frac{U^2}{b^2}h''+S\frac{U^2}{b^2}\alpha''+m{\omega_h}^2h=-L(\tau) \quad (2.17)$$

$$S\frac{U^2}{b^2}h''+I_\alpha\frac{U^2}{b^2}\alpha''+I_\alpha{\omega_\alpha}^2\alpha=M(\tau), \quad (2.18)$$

where $\omega_h=\sqrt{k_h/m}$ and $\omega_\alpha=\sqrt{k_\alpha/I_\alpha}$.

2.3 Classical flutter prediction

The equations that have been derived model a self-excited system. Flutter occurs when the state space model becomes unstable. Since the equations are linear, the stability is checked by calculating the poles of the system. This is simple to do once the model has been converted to a state space format, since it becomes an eigenvalue calculation. However, to write the equation in a matrix form, it is important to notice that the Wagner's function introduces some exponential terms in the time domain. Those terms can be dealt with by adding two states to the system, which are usually called the lags. The derivation of the state space model is shown is Appendix A. Once the geometry of the wing is specified, the state space matrix A becomes a function of air density ρ and air speed U only. It is interesting to plot the evolution of the eigenvalues of the A matrix with respect to one of the two variables, the other one remaining constant. Such a plot is called a root locus, and an example is shown in Figure 2-2. In this case, the air speed varies from 200 ft/sec to 500 ft/sec. Notice that the aircraft structure becomes unstable for air speed greater than 429 ft/sec. This instability point, marked with a star on the figure, is the flutter boundary point that we are interested in.

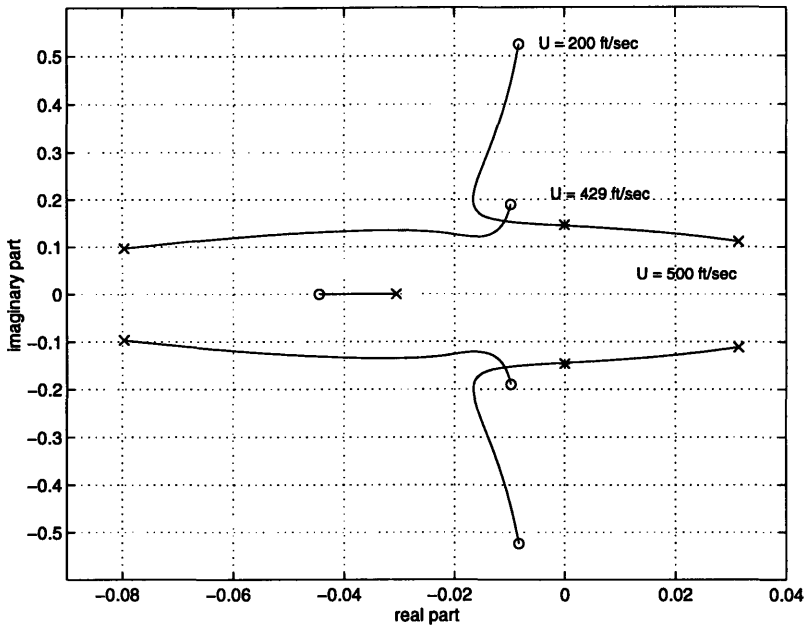


Figure 2-2: Root locus of the flexible aircraft with respect to air speed.

As stated in the introduction, the problem addressed in this thesis is to predict the flutter boundary based on flight data that are necessarily taken in the stable region of the structure. Therefore, it is not possible to obtain the root locus of the flexible structure close to the flutter boundary. The solution is then to extrapolate the results that were obtained at lower air speed or lower air density up to the point of instability. Let us assume, at first, that the poles of the system can be calculated exactly by using some identification techniques. It is then possible to calculate the error that is made by extrapolating the poles to higher speed. To obtain an idea of the accuracy of the results, let us plot the real part as well as the damping ratio (which has more physical meaning) of the pole that is going unstable with respect to speed (Figure 2-3). Notice that the change of behavior of this parameter is very abrupt in the neighborhood of the flutter boundary. This means that the extrapolation will give an accurate answer only when the air speed of the point at which the poles are calculated is very close to the boundary, typically on the right side of the dashed line. In other words, to obtain a reliable answer by extrapolating the damping ratio of the pole, we need to know a priori the answer with an uncertainty of less than 4%. An a priori estimate of the flutter boundary can be obtained by a finite element modeling of the aircraft and aerodynamics. However, the results would usually not be accurate

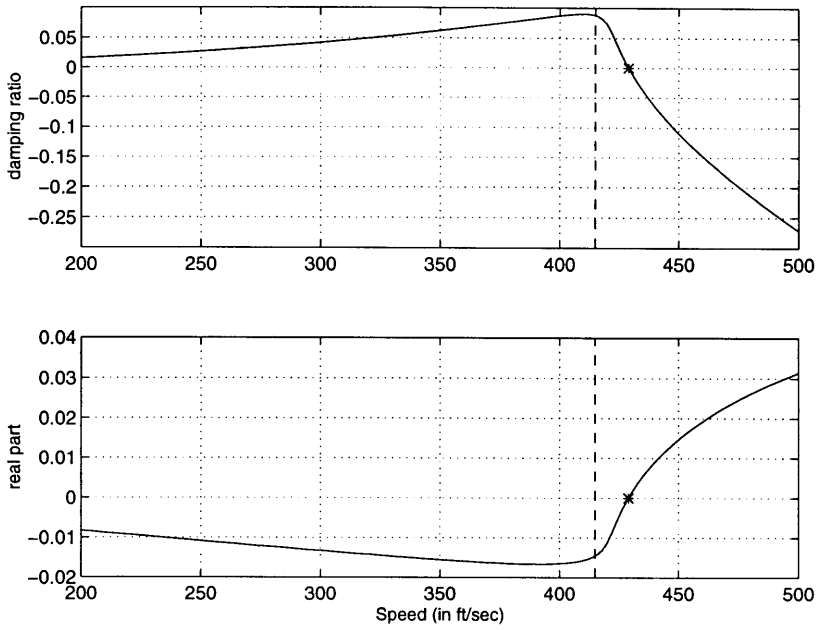


Figure 2-3: Evolution of the damping ratio and the real part of the torsion mode of the aircraft with respect to air speed.

enough to allow the use of such a method. The noise was not taken into account up to this point of the analysis. In real life, noise can never be avoided, providing even more uncertainty in the results. This is amplified by the fact that we need to evaluate the derivative of a curve to predict flutter, which is usually very sensitive to noise.

It was shown, in this section that extrapolating the damping ratio with respect to the air speed or the air density is accurate only when data is available close to the flutter boundary. In practice, this constraint is very limiting because flying close to the instability region is very dangerous for the pilot. Another problem is that the flutter boundary is originally unknown, so in order to fly close to it, it may require many expensive flight tests. An alternate method, explained in Chapter 4, is necessary to solve this problem.

Chapter 3

Classical identification techniques

In this thesis, the goal is to predict flutter boundary by relying heavily on recorded data. The general idea is to obtain an accurate state space model of the structural dynamics of the aircraft that is in agreement with both flutter theory and experimental data. Therefore, an essential step of the procedure is system identification. This topic has been extensively studied in the past and a fairly comprehensive survey of existing methods can be found in [23]. The main techniques are described briefly in this chapter but special attention is devoted to subspace identification. Some interesting contributions on data sets combination are also described in detail, illustrated by a series of applications.

3.1 Parameter estimation method

The first step of a parameter estimation method is to define a model which is in accordance with the physics of the system to identify. Many different models have been proposed in the past. The most popular one is the Auto Regressive Model (ARX) which is presented briefly. The general ARX model has the following form:

$$y(t) = \sum_{k=1}^{n_a} A_k y(t-k) + \sum_{k=1}^{n_b} B_k u(t-k) + n(t), \tag{3.1}$$

where y is the output, u the input and n the noise. n_a and n_b are two integers that respectively represent the order and the number of zeros of the model. Those two numbers are chosen by the user based on physical knowledge on the system. However, n_b should be smaller than n_a to get a strictly proper system. The other parameters, A_k and B_k, are constants to evaluate through the identification method. Writing Equation (3.1) in a matrix form leads to

$$\begin{bmatrix} y_1 & \cdots & y_{n_a} & u_{n_a-n_b+1} & \cdots & u_{n_a} \\ \vdots & & & & \vdots & \\ y_p & \cdots & y_{p-1+n_a} & u_{p+n_a-n_b} & \cdots & u_{p-1+n_a} \end{bmatrix} \begin{bmatrix} A_1 \\ \vdots \\ A_{n_a} \\ B_1 \\ \vdots \\ B_{n_b} \end{bmatrix} = \begin{bmatrix} y_{n_a+1} - n_{n_a+1} \\ \vdots \\ y_{p+n_a} - n_{p+n_a} \end{bmatrix} \tag{3.2}$$

where p is an index for the number of data samples. Equation (3.2) is a set of linear equations in A_k and B_k that are solved through a least square method.

$$\begin{bmatrix} A_1 \\ \vdots \\ A_{n_a} \\ B_1 \\ \vdots \\ B_{n_b} \end{bmatrix} = \begin{bmatrix} y_1 & \cdots & y_{n_a} & u_{n_a-n_b+1} & \cdots & u_{n_a} \\ \vdots & & & & \vdots & \\ y_p & \cdots & y_{p-1+n_a} & u_{p+n_a-n_b} & \cdots & u_{p-1+n_a} \end{bmatrix}^{\dagger} \begin{bmatrix} y_{n_a+1} - n_{n_a+1} \\ \vdots \\ y_{p+n_a} - n_{p+n_a} \end{bmatrix} \tag{3.3}$$

If we assume that the noise is white, with zero mean and a variance of ${\sigma_n}^2$, we can note that the estimate is unbiased, since the expected value of the error e equals

$$E(e) = \begin{bmatrix} y_1 & \cdots & y_{n_a} & u_{n_a-n_b+1} & \cdots & u_{n_a} \\ \vdots & & & & \vdots & \\ y_p & \cdots & y_{p-1+n_a} & u_{p+n_a-n_b} & \cdots & u_{p-1+n_a} \end{bmatrix}^{\dagger} E(n) = 0 \tag{3.4}$$

Denoting

$$Q = \begin{bmatrix} y_1 & \cdots & y_{n_a} & u_{n_a-n_b+1} & \cdots & u_{n_a} \\ \vdots & & & & \vdots & \\ y_p & \cdots & y_{p-1+n_a} & u_{p+n_a-n_b} & \cdots & u_{p-1+n_a} \end{bmatrix}^{\dagger}, \tag{3.5}$$

The variance σ^2 of the estimate is

$$\sigma^2 = E(ee^T) = \sigma^2 QQ^T \tag{3.6}$$

Note that the variance of the estimate is linked to the singular values of the matrix Q. The variance can be decreased by increasing the amplitude of the input signal, which means adding more energy to the system. This assertion is true as long as non-linearities in the system can be neglected which implies magnitudes of the input signal to be small. Therefore, there is a trade off.

Other types of model can be found in the literature such as the Auto Regressive Moving Average Model (ARMAX), Output Error models (OE), Finite Impulse Response models (FIR) or the Box-Jenkins models (BJ). A more comprehensive list can be found in [24, 23]. Each model assesses some properties on the system such the number of modes or the properties of the noise. For example, the ARMAX model has the following structure

$$y(t) = \sum_{k=1}^{n_a} A_k y(t-k) + \sum_{k=1}^{n_b} B_k u(t-k) + \sum_{k=1}^{n_c} C_k n(t), \tag{3.7}$$

where some dynamics are added to the noise.

Other types of model assume that the input is an unknown noise having some known properties. For example, the Auto Regressive Model (AR) is the following

$$y(t) = \sum_{k=1}^{n_a} A_k y(t-k) + n(t), \tag{3.8}$$

Methods to identity such models are usually called prediction methods and those problems are addressed in [25, 2].

3.2 Subspace identification

Subspace identification methods have been initiated by the works of Kung [19], and Juang and Pappa [17]. A variety of new methods have emerged ([31], [28], [4] and [22]) for identifying a system in the time domain, [32] for systems with stochastic input, and also [26] in the frequency domain. Efficient numerical procedures using the structures of Hankel and Toeplitz matrices, saves computational time and storage, increasing the performances of such algorithms [5]. All those methods are based on the same basic principle, presented in this section through a simple subspace identification algorithm.

3.2.1 Notations

The goal of subspace identification is to find a linear, time invariant, finite dimensional state space realization

$$x_{k+1} = Ax_k + Bu_k \tag{3.9}$$

$$y_k = Cx_k + Du_k,$$

where $A \in \Re^{n\times n}$, $B \in \Re^{n\times m}$, $C \in \Re^{l\times n}$, $D \in \Re^{l\times m}$, based on the knowledge of specific sequences $u = [u_1, ..., u_p]$, $y = [y_1, ..., y_p]$.

The following notation is used:

The block Hankel input and output matrices are defined as

$$Y_h(k,i,j) = \begin{bmatrix} y_k & y_{k+1} & \cdots & y_{k+j-1} \\ y_{k+1} & y_{k+2} & \cdots & y_{k+j} \\ \cdots & \cdots & \cdots & \cdots \\ y_{k+i-1} & y_{k+i} & \cdots & y_{k+j+i-2} \end{bmatrix}$$

and

$$U_h(k,i,j) = \begin{bmatrix} u_k & u_{k+1} & \cdots & u_{k+j-1} \\ u_{k+1} & u_{k+2} & \cdots & u_{k+j} \\ \cdots & \cdots & \cdots & \cdots \\ u_{k+i-1} & u_{k+i} & \cdots & u_{k+j+i-2} \end{bmatrix}.$$

We also introduce the extended observability matrix

$$\Gamma = \begin{bmatrix} C \\ CA \\ \cdots \\ CA^{i-1} \end{bmatrix},$$

the lower block triangular Toeplitz matrix

$$H_{tl} = \begin{bmatrix} D & 0 & 0 & \cdots & 0 \\ CB & D & 0 & \cdots & 0 \\ CAB & CB & D & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ CA^{i-2}B & CA^{i-3}B & CA^{i-4}B & \cdots & D \end{bmatrix},$$

and the state matrix

$$X = \begin{bmatrix} x_k & x_{k+1} & \cdots & x_{k+j-1} \end{bmatrix}.$$

This notation leads to the following representation of the input output history:

$$Y_h(k,i,j) = \Gamma X + H_{tl} U_h. \tag{3.10}$$

3.2.2 Step by step procedure

The step by step procedure of a subspace identification algorithm with one data set is now explained through the example of the deterministic identification (i.e. no noise is corrupting the data).

Step 1: find a matrix P that satisfies an equation of the form

$$P = \Gamma Q, \tag{3.11}$$

where Γ is the extended observability matrix and such that rank(P)=rank(Γ)=n.
In practice, the existence of noise makes it impossible to obtain equation (3.11) exactly. Any subspace method extracts a matrix P from the input to output data that is optimal in the sense defined by the method: the specific solution depends mainly on the noise assumption. Depending on the subspace method that is chosen, different computations of this matrix P are possible, all leading to different results.
In the case of a deterministic system, P can be found by post multiplying equation (3.10) by a matrix ${U_h}^{\perp}$ that satisfies $U_h {U_h}^{\perp} = 0$. We then obtain $P = Y_h {U_h}^{\perp}$. However, the rank of the matrix P may not be equal to the order of the system. This phenomenon is known as rank cancellation and its probability of occurring decreases when the number of rows in Y_h increases.

Step2: perform a singular value decomposition of P

$$P = USV,$$

where $S = \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix}$ *and* $U = (U_1\ U_2)$ *such that* U_1 *is the first n columns of* U.
Note that S_1 is an $n \times n$ matrix. With Equation (3.11), we can see that there must exist a full rank $n \times n$ matrix T such that

$$U_1 = \Gamma T.$$

Let us now use the following notation: if M is an $m \times n$ matrix, $\overline{M}$ (resp. $\underline{M}$)

will be the matrix with a reduced number of rows, obtained from M by omitting the first (resp. last) l rows, where l is the number of outputs of the system.

Step 3: Evaluate A and C as follow: $A = \overline{U_1}^{\dagger}\underline{U_1}$ and C is equal to the first block of U_1, where $\overline{U_1}^{\dagger}$ denotes the pseudo-inverse of $\overline{U_1}$.

Using the structure of the extended observability matrix, it is clear that

$$\overline{\Gamma} = \underline{\Gamma}A$$

$$\underline{U_1} = \underline{\Gamma}T \ , \ \overline{U_1} = \overline{\Gamma}T$$

$$\underline{U_1}T^{-1} = \overline{U_1}T^{-1}A.$$

This can also be written as

$$\underline{U_1} = \overline{U_1}\Psi \ , \ \Psi = T^{-1}AT.$$

Thus, Ψ is a matrix similar to A, which is what we wanted originally.

Step 4: Use a least square method to compute B and D.

We can pre multiply equation (3.10) by $\Gamma^{\perp}$ such that $\Gamma^{\perp}\Gamma = 0$, and post multiply it by the pseudo-inverse of U_h. By using the structure of the matrix H_{tl}, we get

$$\Gamma^{\perp}Y_h{U_h}^{\dagger} = \Gamma^{\perp}\begin{bmatrix} I & 0 \\ 0 & \underline{\Gamma} \end{bmatrix}\begin{bmatrix} D \\ B \end{bmatrix},$$

leading to

$$\begin{bmatrix} D \\ B \end{bmatrix} = (\Gamma^{\perp}\begin{bmatrix} I & 0 \\ 0 & \underline{\Gamma} \end{bmatrix})^{\dagger}\Gamma^{\perp}Y_h{U_h}^{\dagger}.$$

3.3 Multiple data sets in subspace identification

Currently available time-domain subspace identification algorithms assume that plant identification is based on a single experiment, where only a single input to output data

set is available. There are, however, many cases for which data collection cannot be done all at once, and experiments must be segmented possibly over a period of several days, leading to the collection of many data sets all related to the same dynamic system, but with possibly different initial conditions. This is typically the case, for example, when attempting to identify the flexible dynamics of the F18 Systems Research Aircraft (SRA) at NASA Dryden Flight Research Center, where several data sets generated through many flights at the same flight conditions (altitude, Mach number and dynamic pressure) are available.

The idea of combining data sets into one single identification method is not new (see [20]). However, it has never been implemented on subspace identification methods. In this section, it is shown how such an algorithm may be readily adapted to handle multiple data sets.

3.3.1 Motivational example

Before we start presenting the algorithm with multiple data sets, an example is first described to show that naive concatenation of the data sets leads to severely degraded performance. Results are compared with a method described in Section 3.3.2 and in [7] which recovers the original performance of subspace algorithms.

The system is an 8^{th} order discrete time system with one input and two outputs, whose state space representation can be found in Appendix B. The system has been excited separately by two sets of linear frequency sweeps. The choice of such inputs has been motivated by some practical concerns since frequency sweeps were the only available excitations at our disposal to identify the structural dynamics of the F18-SRA. The following formula for the inputs has been used from $k = 100$ to 3000, the first 100 points were set to zero:

$$\begin{aligned} e1(k) &= \sin(2\pi(5 + 20k/3000)(k - 100)/3000) \\ e2(k) &= \cos(2\pi(5 + 20k/3000)(k - 100)/3000). \end{aligned}$$

The simulation of this system has been realized for each input and the two data

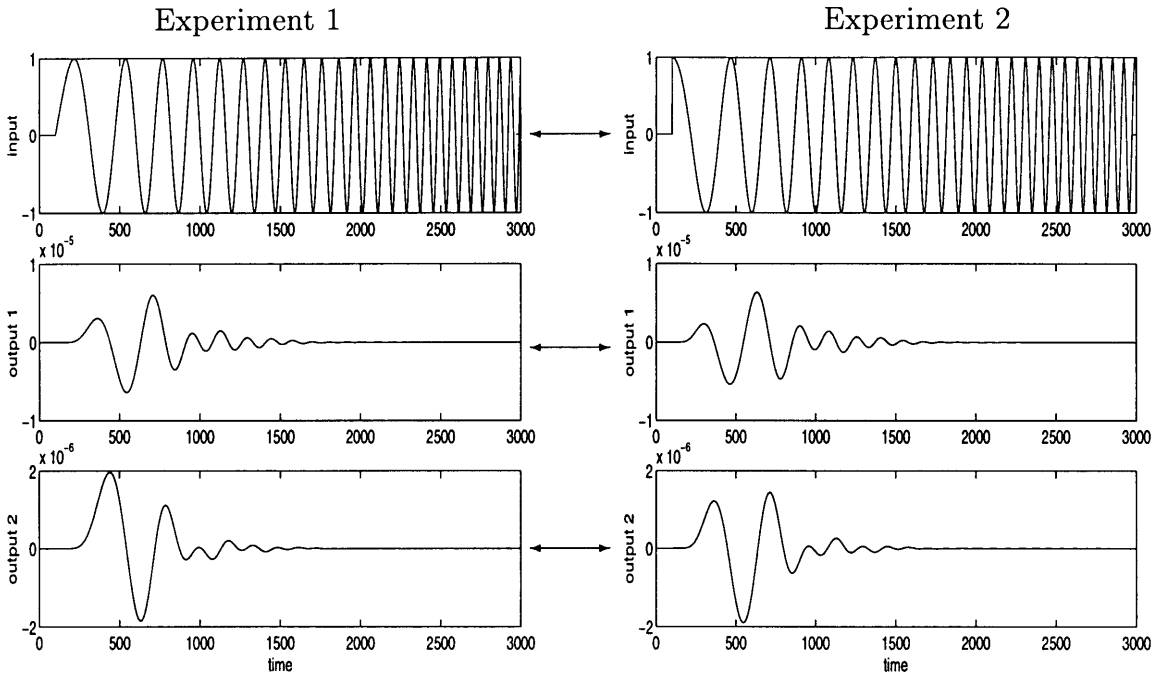


Figure 3-1: Concatenation of two simulations made on a 8^{th} order system with two different inputs and no noise.

sets were concatenated. The plot of the input and outputs can be seen in Figure 3-1. Notice that the discontinuity at the junction of the two data sets is very small. Then, the identification of the system with a subspace algorithm (we used N4SID which is a state of the art method) was performed as if the concatenated data had been recorded from only one experiment. The number of blocks i in the Hankel matrix was set to 14, 15 and 16. For $i = 15$, the original system was perfectly recovered. The problem came when $i = 14$ or 16 was used since some of the eigenvalues have become unstable as seen on Table 3.1. Other values of i have been tested from 10 to 30 and the algorithm failed in about 70 % of the cases. Even though the identification was accurate for some values of i, the issue remains; the user has no way to discriminate between the right answer and the wrong one.

On Figure 3.3.1, the simulation of the system with the input *concatenated* is realized, and the outputs of this *single* experiment are plotted. By comparing those outputs to the one shown on Figure 3-1, it can be noticed that the difference between the two tests is very small. However, when applied to these data, N4SID recovered the

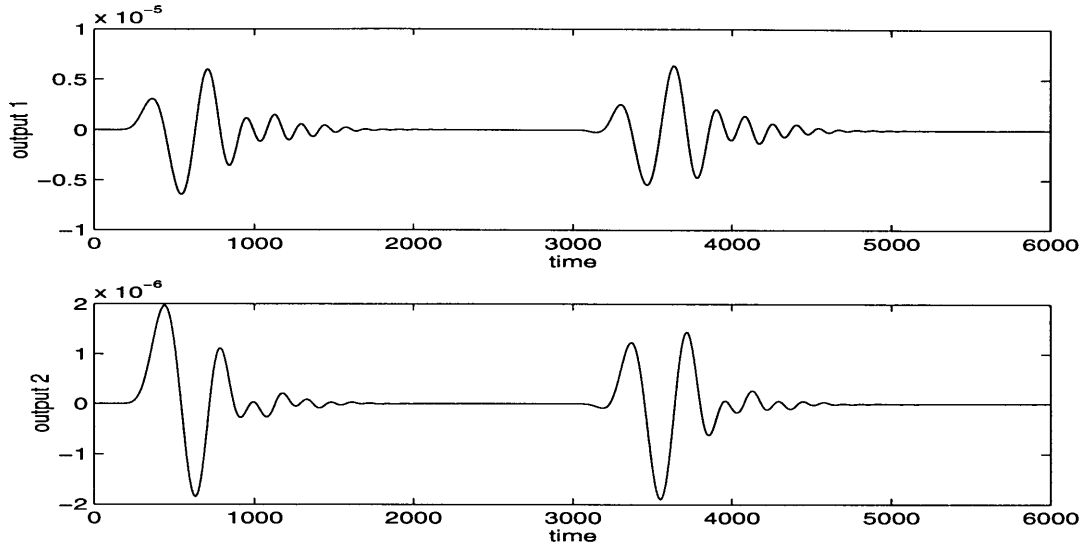


Figure 3-2: Simulation made with the same system as in Figure 3.1 but with the concatenated input.

original eigenvalues	eigenvalues calculated by concatenating the two sets		eigenvalues calculated with this new algorithm
	$i = 14$	$i = 16$	
0.9893 + 0.0396i	.9977+.0100i	1.0133 + 0.0614i	0.9893 + 0.0396i
0.9893 - 0.0396i	.9977-.0100i	1.0133 - 0.0614i	0.9893 - 0.0396i
0.9799 + 0.0245i	.9960+.0200i	0.9969 + 0.0377i	0.9799 + 0.0245i
0.9799 - 0.0245i	.9960-.0200i	0.9969 - 0.0377i	0.9799 - 0.0245i
0.9949 + 0.0149i	.9944+.0386i	0.9985 + 0.0098i	0.9949 + 0.0149i
0.9949 - 0.0149i	.9944-.0386i	0.9985 - 0.0098i	0.9949 - 0.0149i
0.9753	.9454+.1431i	0.9976 + 0.0195i	0.9754
0.9851	.9454-.1431i	0.9976 - 0.0195i	0.9850

Table 3.1: Eigenvalues of the identified model

right eigenvalues regardless of i. This shows that the identification procedure is very sensitive to data corruption.

To show that this problem does not come from the kind of input that has been chosen, the system was identified with each data sets separately. The original system was recovered with any i that we picked for both data sets.

N4SID was then adapted to handle multiple data sets, using the algorithm described in the next section. This modified version was used on the same numerical data. As shown in table 3.1, the result of this identification was very accurate. The eigenvalues were fitted with an error lower than 0.1%.

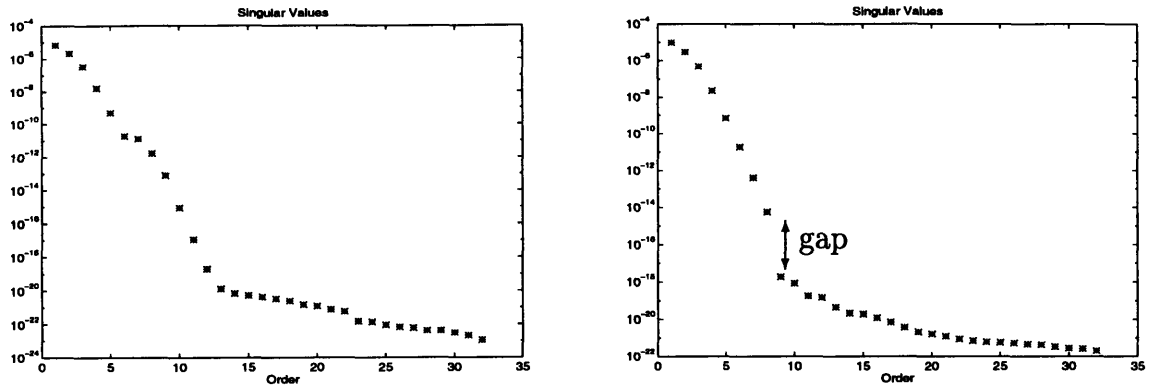


Figure 3-3: Singular values to estimate the order of the system. The left picture happens when concatenating the data, the right one is with the new scheme.

The question of determining the order of the system is also a major issue in identification methods. In practice, the order is also unknown and needs to be determined. In many subspace identification algorithms, the singular values of the matrix P (step 1) are plotted and the user must decide the order of the system. If there is a jump in the amplitude of the singular values, the order is determined by the number of singular values to the left of this jump. If there is no detectable jump, then the user must guess the system's order, based on his or her knowledge of the system. Figure 6 shows the plots that are obtained using both procedures with 16 blocks in the Hankel matrix ($i = 16$). Notice that it is not obvious to determine the order of the system when the data has been concatenated. On the other hand, there is a gap of 3 orders of magnitude for the improved procedure.

3.3.2 Algorithm

We will now assume that we have collected two data sets (the generalization to n data sets is very simple and is omitted for notation purposes), $u_1(k)$, $y_1(k)$ and $u_2(k)$, $y_2(k)$ and the following equations are satisfied

$$Y_1 = \Gamma X_1 + HU_1 \tag{3.12}$$

$$Y_2 = \Gamma X_2 + HU_2.$$

Let us now explain how the original algorithm has to be modified in order to handle multiple data sets.

Step 1: Find two matrices P_1 and P_2 that satisfy $P_i = \Gamma Q_i$, for $i = 1, 2$, where Γ is the extended observability matrix.

Actually, this step is similar to the first step of the initial algorithm, but we need to perform it for each data set. For example, if we want to use the noise free method, we should proceed as follow

$$P_1 = Y_1 U_1{}^{\perp} = \Gamma(X_1 U_1{}^{\perp})$$

$$P_2 = Y_2 U_2{}^{\perp} = \Gamma(X_2 U_2{}^{\perp}).$$

The main modification of the algorithm is to compute an additional step at this point.

Step 1bis: Compute the matrix $\Phi = [P_1\ P_2]$.

This matrix Φ satisfies

$$\Phi = \Gamma[Q_1\ Q_2],$$

which is exactly the same property as the matrix P of the first steps of the original algorithm.

The steps 2 to 4 are exactly the same as in the original algorithm, where the matrix Φ replaces the matrix P.

3.3.3 Remarks

If we append the two data sets at the beginning of the experiment and use the single data set algorithm, the Hankel matrix Y_h will have some columns that have no physical meaning. This is because, at the junction of the two data sets, some columns will contain some data from both experiments as shown in Equation (3.13) and Equation (3.10) would not be satisfied anymore.

$$\begin{bmatrix} y_1(1) & \cdots & y_1(p-i+2) & \cdots & y_1(p) & y_2(1) & \cdots & y_2(q) \\ y_1(2) & \cdots & y_1(p-i+3) & \cdots & y_2(1) & y_2(2) & \cdots & y_2(q+1) \\ \vdots & & & & & & & \vdots \\ y_1(i) & \cdots & \underbrace{y_2(1) \quad \cdots \quad y_2(i-1) \quad y_2(i)}_{\text{No physical meaning}} & & & & \cdots & y_2(q+i-1) \end{bmatrix} \tag{3.13}$$

If the classical algorithm were used, those columns would be considered as part of the dynamics of the system. On the other hand, the proposed method avoids this problem by removing those undesirable columns. The algorithm treats those data sets in parallel, and concatenates them only when performing a least square fit so that only the real dynamics are kept. Therefore, the statistical properties such as the bias or the variance of the estimator of the state space model are carried over.
Only deterministic subspace identification has been detailed in this thesis because it is the easiest one to understand. However, this method can be applied to more sophisticated algorithms such as N4SID.

3.3.4 Examples

An academic case

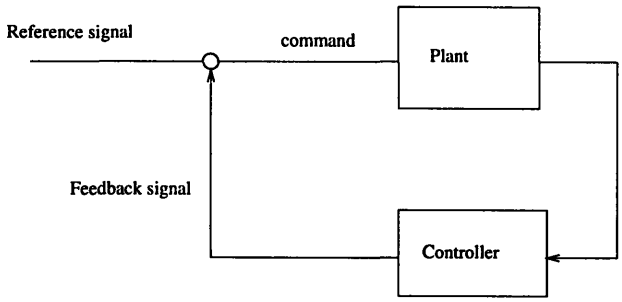
Let us start our series of examples with a very academic one. The system that was chosen is a 4^{th} order system with two real poles at 0.5 and -0.25, and two complex conjugate poles at $0.3e^{\pm i\pi/3}$. The matrices B and C are chosen so that the system is fully observable and controllable. The simulation was driven by a known input which was generated by a white noise process. Some additional unknown white noise was also added to the output. Each simulation lasted exactly 20 samples. The identification was then made with the deterministic subspace algorithm and with a set of independent experiments. The value of i in the algorithm was always set to 10 and the identified system order to 4. The evolution of the eigenvalues with respect to the number of experiments is plotted on Figure 3-4. It appears that all

the eigenvalues converge to the actual value of the plant. Of course, since noise is corrupting the signals the convergence is not monotonic, but the variance of the error tends to decrease.

The same data was then concatenated and treated as one *single* experiment with the same simple subspace identification and the eigenvalues did not converge at all. Actually, all the eigenvalues were identified as being complex.

An interesting application of this algorithm is to identify unstable systems. Indeed, an unstable system cannot by driven for a long period of time in practice because saturation will occur very quickly. Therefore, only a few valid samples for each experiment are available.

To avoid the saturation, the identification could be done in closed loop, using the following standard set up:



However, to stabilize an unstable plant, the feedback signal usually has a very high amplitude compared to the reference signal. This means that the command is almost equal to the feedback. Therefore, the spectrum of the command does not cover all the frequency range. This property usually leads to poor identification performance.

To show that the proposed method also works for an unstable system, the same experiment as before was generated with the mode at 0.5 switched to 1.2. The results are plotted in Figure 3-5, and we notice that the modes are still very well identified. Note also that the unstable pole was identified after the first experiment, but the multiple data set algorithm showed improvements for the other eigenvalue.

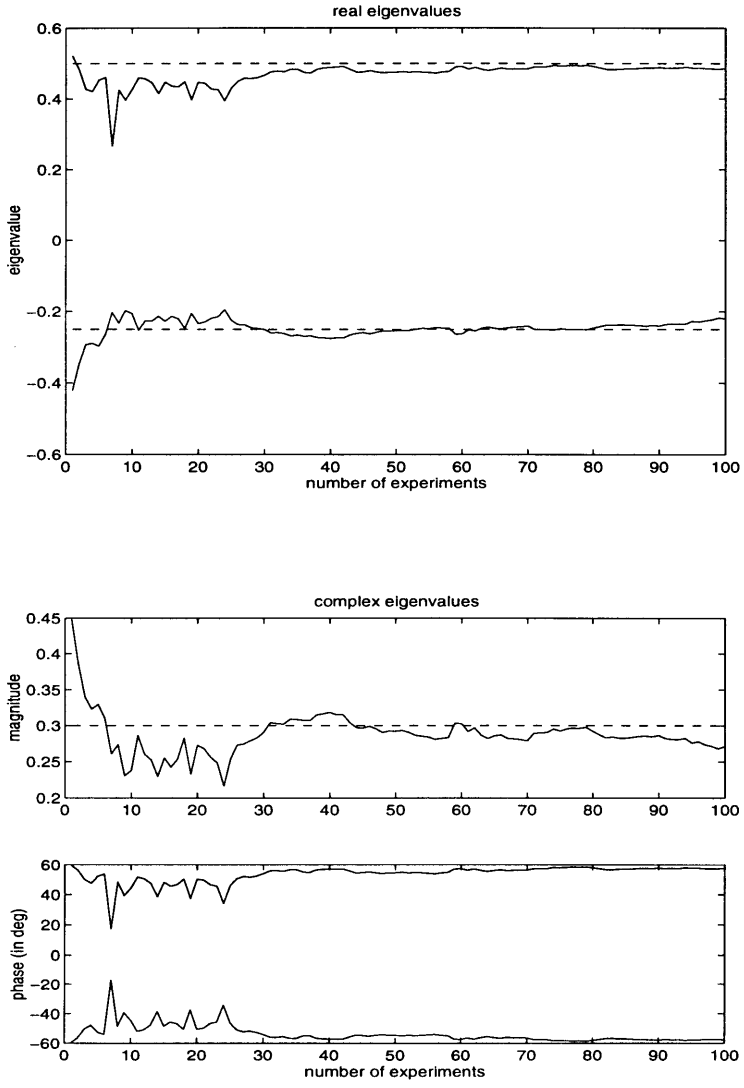


Figure 3-4: Identified eigenvalues with respect to the number of experiments for a stable system.

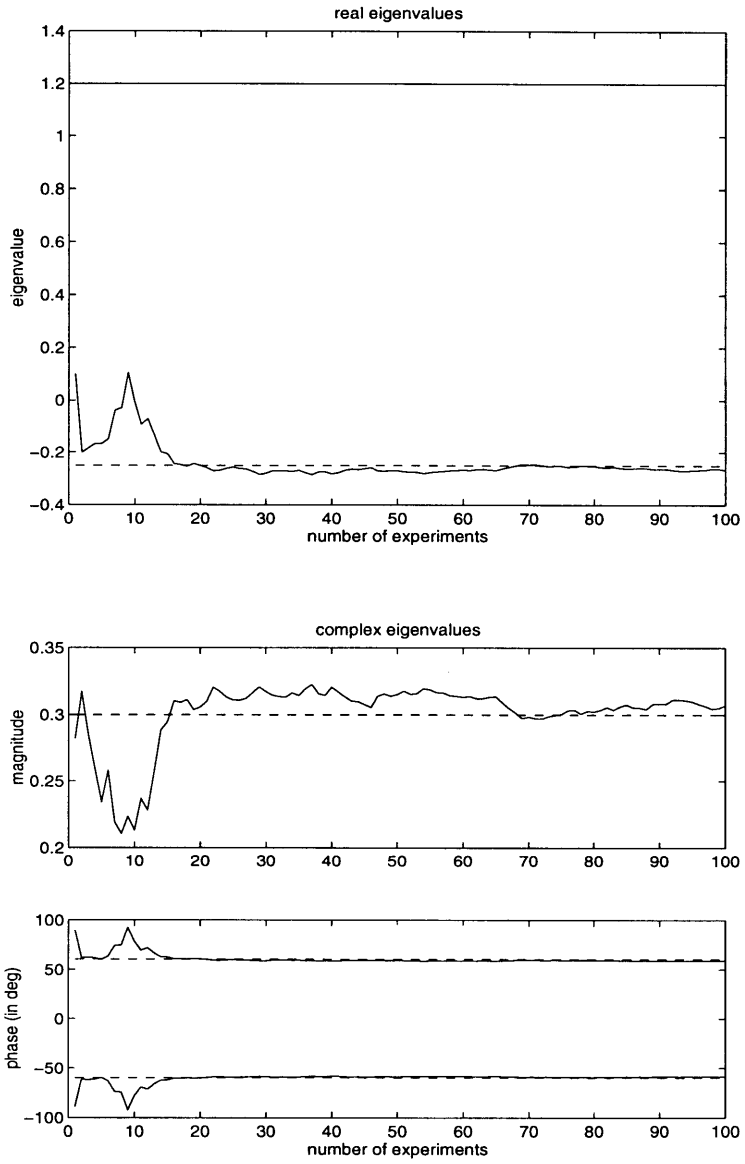


Figure 3-5: Identified eigenvalues for an unstable system.

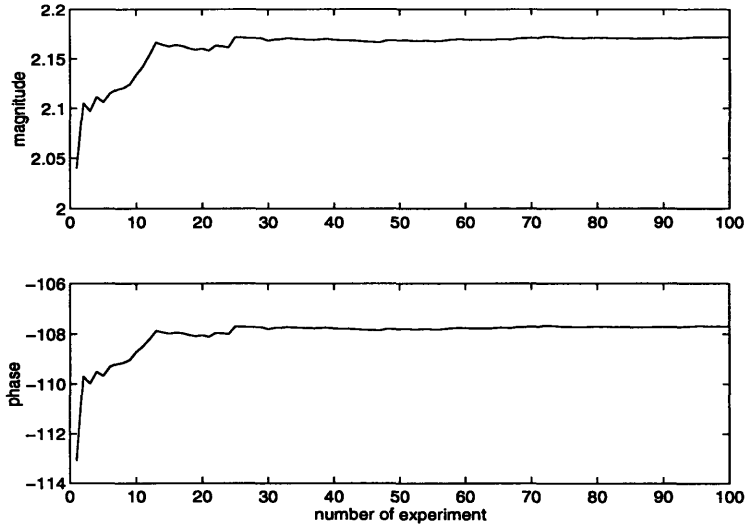


Figure 3-6: Convergence of the short period eigenvalue for the steady state case.

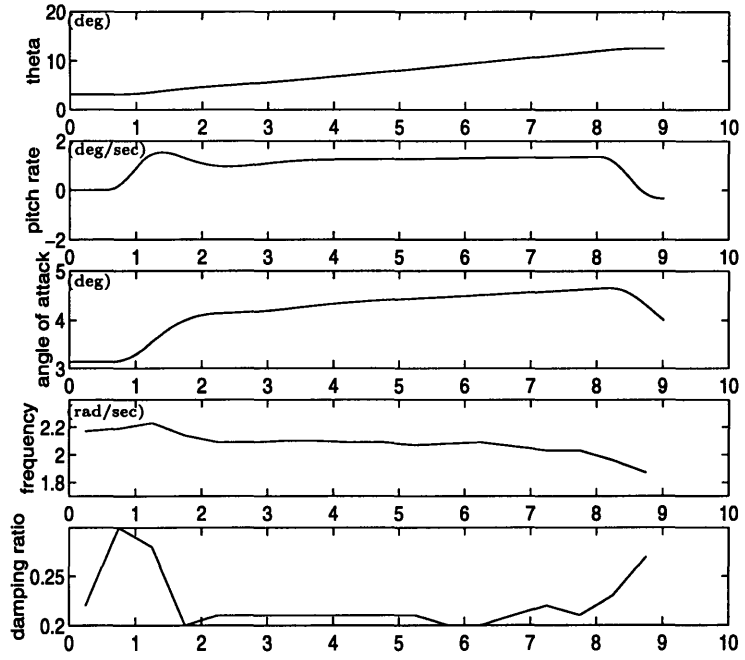


Figure 3-7: Evolution of the short period eigenvalue with respect to time in a pull up maneuver.

F18 longitudinal dynamics

A more practical application of this tool is to identify a linearized model of a system at a point which is not an equilibrium point. In other words, we are looking at the linearized dynamics around an arbitrary trajectory. Its main application would be to realize an optimized gain scheduling for the control system of this plant [27]. The linearized dynamics can be provided by a direct linearization of the equations of motion. However, this is a long and time-consuming process, and subspace identification methods can be used to solve this problem more efficiently.

In the case of an F18, the linearized dynamics around that trajectory will be time-varying, but we expect that they will be a function of the state variables only. This assumption would simply mean that fuel variation is neglected. In such a case, it is clear that it is impossible to stay for a long period of time at the same operating point, since either pitch, yaw or roll rates would not be identically zero. If we try to identify the plant with with only one experiment, the amount of data would be too small to obtain a reliable model. Therefore, we need to collect data from different experiments in order to obtain a reliable identified plant.

An accurate simulator of the F18, developed in cooperation between NASA Dryden Flight Research Center and MIT, was used to generate an example. Indeed, a pull up maneuver was simulated and some white noise excitation was added to the horizontal tail. The white noise amplitude was set to be small so that the nominal trajectory would not be affected. The initial flight point was chosen at Mach 0.6 and 15000 feet of elevation. To show that subspace identification methods with multiple data sets can be applied to an aircraft, the experiment was first run at a steady level flight with an excitation that lasted .5 seconds. The input was added directly to the horizontal tail and the simulation was done in closed loop with the actual control law in the feedback path. However, only the open loop plant was identified by collecting data from the horizontal tail and from the longitudinal states of the airplane (vertical speed, forward speed, pitch and pitch rate). The nominal values of those states were subtracted and the subspace identification method was then run with the perturbed

values. As a matter of fact, the phugoid could not be identified by this method. The reason is that this mode has a very low frequency (.1 Hz), so by taking runs of experiments of only 0.5 seconds (1/20 of a period) this mode cannot be observed properly. However, the short period eigenvalue has converged to a value which is very close to the one estimated by NASA Dryden with an other simulator, as shown in Figure 3-6.

Then, the simulation was run with a command input of 1.5 degrees/sec of pitch rate. At the initial time, the aircraft still was at Mach 0.6 and at 15000 feet of elevation. A linear model of the plant was then identified at every step of the maneuver. Here again, the excitation was added directly on the horizontal tail for a period of 0.5 seconds around the operating point of interest. No linear models of the F18 around a pull up maneuver have been found in the literature. Therefore, the results cannot be compared to actual values but still, they are plotted in Figure 3-7. The conclusion that can be drawn is that the multi-data sets algorithm provides such information that can be used, in the future, for more efficient control law design.

3.4 Conclusion

All the methods that were presented in this chapter have been used extensively in the past and have given very reliable results. However, a major constraint remains: there is not enough structure in the state space model obtained with such identification techniques. In other words, those methods give a very good representation of the dynamics of the system at each flight conditions, but it is very hard to establish a correlation between the different flight conditions. This is a major constraint for flutter boundary prediction since this is the information that we are interested in.

Those identification methods could however be applied to estimate a transfer function. A different method is presented in the next chapter but it relies on the type of input that was used in the experiments. In the future, if different excitation signals are used such as white noise, subspace identification may find some interesting application in a flutter boundary determination problem.

Chapter 4

A new methodology

As stated in Chapter 3, the major problem of the previous identification method was that the state space model had no structure that could be carried over from one flight condition to another. The main reason is that the physics was not represented well enough in the model resulting from classical system identification methods. The solution was found by using an accurate model of the structural dynamics of an aircraft close to fluttering flight conditions. The detailed approach is illustrated by the flowchart shown in Figure 4-1. As one can see, three essential steps are involved in the proposed flutter boundary estimation procedure. The first step gives an estimate of the transfer function of the system and relies on the *a priori* knowledge of the exciting signal. The second step takes the transfer function estimate and fits a physical model to it. The last step uses the parameterized model and projects it to the flutter boundary by varying Mach number and dynamic pressure.

4.1 Transfer function estimation

Classical transfer function estimation techniques, developed in [30], as well as the identification methods described in Chapter 3 have been tried on both simulated and experimental data. The results of those methods were not satisfactory enough for the purpose of flutter boundary estimation because the properties of the input signal

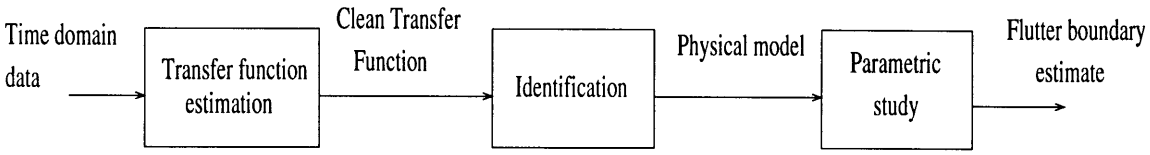


Figure 4-1: Flowchart of the methodology

were not used effectively. Indeed, the input was a frequency sweep

$$u(t) = U_0 \sin(2\pi g(t)).$$

This sweep is characterized by the properties of the function g. The derivative of g with respect to time $\frac{dg}{dt}$ is called the instantaneous frequency f. In the case of a flutter problem, the frequency sweep is either linear, which means that

$$\frac{dg}{dt} = kt, \tag{4.1}$$

or logarithmic

$$\frac{dg}{dt} = k \log(t). \tag{4.2}$$

Work on transfer function estimation of systems excited by a linear frequency sweep was found in [6]. However, the limitation was that the slope k of the frequency sweep had to be very slow in order to apply this method. This limitation is a major constraint in aeronautics, given the high cost of flight tests.

New ideas emerging from wavelet analysis [35] gave rise to a new transfer function estimation method based on time-frequency analysis. Moreover, the input signal has an appreciable power spectral density at a frequency ω_0 only during a short period

of time. Assuming that the system is linear, the output should also have a non zero power spectral density at ω_0 only around this time. Therefore, the transfer function estimate should rely only on data from this time interval.

4.1.1 Principle of the estimation

A transfer function is supposed to be a continuous function of frequencies which it is impossible to deal with since there would be an infinite number of frequencies. It is thus necessary to discretize the frequency axis and calculate the transfer function at only a finite amount of points. The choice of those frequencies will be addressed in Section 4.1.2. In the rest of this section, the interest is focused on estimating the transfer function at one given frequency ω_0.

In this method, all the outputs are processed separately so that the system can be considered as multiple input, single output. In the first step of the estimation, all the inputs and the output are filtered by the same band pass filter centered around the frequency ω_0 . Filtering the signals does not alter the identification at all. Indeed, if the filter F is linear time invariant (LTI), this operation can be seen as a multiplication by the following scalar matrix:

$$\begin{bmatrix} F & & 0 \\ & \ddots & \\ 0 & & F \end{bmatrix}$$

in which the number of rows is equal to m, the number of inputs. If we note $\tilde{U}$ and $\tilde{Y}$ the filtered input and output, it follows that:

$$\tilde{Y} = FY = FGU = G \begin{bmatrix} F & & 0 \\ & \ddots & \\ 0 & & F \end{bmatrix} U = G\tilde{U}$$

This means that if the filtered input excites the system, the output will be the filtered output. For this reason, in the rest of this section, all the notations related to the

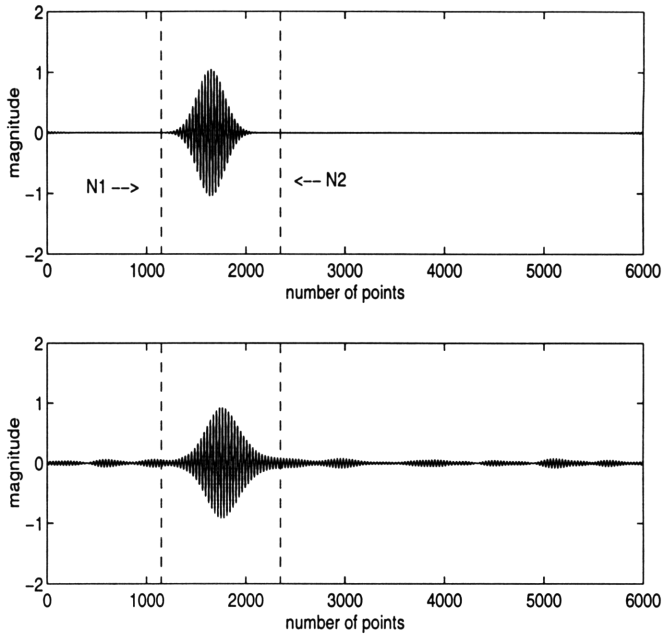


Figure 4-2: Choice of the parameters N_1 and N_2 of the input and output signals

inputs and output will represent the filtered signals.

Now, a time window (from a sample time noted N_1 to a sample time noted N_2 for the input and the output) need to be selected. Physically, this time window corresponds to the time when the input excites the system at the frequency ω_0. It also represents the time when the input signal is actual information and not noise. The choices for the parameters N_1 and N_2 will be discussed in the next section.

The figure 4-2 illustrates this choice. The upper plot correspond to the input signal and the lower plot, to the output. Note that the output signal outside the time window has a significant magnitude. Since there is no energy in the input at this time and this frequency, properties of linear systems reveals that the energy in the output signal outside the interval $[N_1\ N_2]$ just comes from noise. Therefore, only the part of the signal with accurate information (inside $[N_1\ N_2]$) will be kept.

We are now able to compute the transfer function estimate at this frequency: let

$$U_l = \sum_{k=N_1}^{N_2} u_l(k) e^{i\omega_0 k}$$

where u_l represents the l^{th} input of the system. Define also

$$Y = \sum_{k=N_1}^{N_2} y_l(k) e^{i\omega_0 k}.$$

If we assume that there is no noise then

$$Y = \sum_{l=1}^{m} G_l U_l$$

where G_l is the estimate of the transfer function from the l^{th} input to the output. However, this equation is true only in two cases. First, when the system is at steady state during the whole time window and second, when the system is at rest at the beginning of the time window and the time window is long enough so that the system is assumed to be at rest at the end of the time window also.
If we reproduce this procedure for many experiments, we obtain a set of equalities

$$Y_k = [U_{kl}] G_l$$

where U_{kl} correspond to the Fourier transform of the l^{th} input of the k^{th} experiment. If there are enough independent experiments, the matrix of U_{kl} will be of rank m. If so, a least-square solution will give a transfer function estimate

$$G_l = [U_{kl}]^{\dagger} Y_k \tag{4.3}$$

where $[U_{kl}]^{\dagger}$ represents the pseudo-inverse of U_{kl}. In a flutter problem, there are usually two exciters (one on each wing) and a set of symmetric and asymmetric excitations is performed. This provides two linearly independent equations so U_{kl} will be full rank.

Note that if the filter F is an ideal band pass filter, filtering the signal does not affect at all the estimate since the Fourier transform of the input and the output signals should be the same at the frequency of interest. However, this filtering is helpful to determine the length of the time window to consider.

4.1.2 Resolution issues

For practical purposes, it is necessary to determine the length of the time window Δt and the resolution Δf that can be obtained in the transfer function. Of course, it would be interesting to minimize those two parameters. However, a theoretical boundary is defined by the Heinsenberg inequality [35].

$$\Delta t \Delta f > 2. \tag{4.4}$$

Also, the slope of the chirp gives another relation between time and frequency resolution. At this point, we need to make a distinction between the linear and the logarithmic sweeps.

In the case of a linear sweep,

$$\Delta f = k \Delta t. \tag{4.5}$$

Combining this equation with the Heinsenberg inequality (4.4), we obtain

$$\Delta t = \sqrt{2/k},\ \Delta f = \sqrt{2k}. \tag{4.6}$$

In the case of a logarithmic sweep, Equation (4.5) becomes

$$\Delta f = \frac{k}{t} \Delta t. \tag{4.7}$$

Therefore, the time and frequency resolutions become of function of time (and frequency due to Equation (4.2))

$$\Delta t = \sqrt{2t/t},\ \Delta f = \sqrt{2k/t}. \tag{4.8}$$

Of course, those results have an underlying assumption which is that the damping ratio of the modes of the system are not too small so that Equation (4.5) is still satisfied for the output signals.

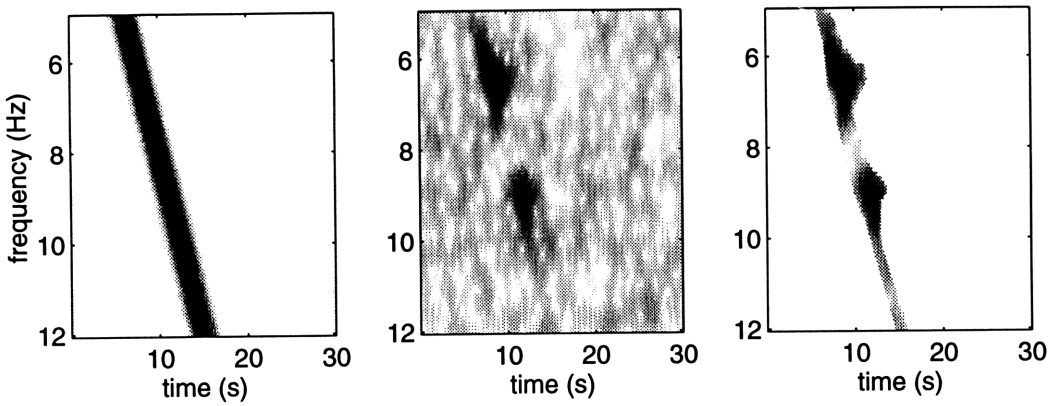


Figure 4-3: Signal representation using the graphical interfacing tool

4.1.3 Graphical interface

Selecting the time window for each signal may be a hard and laborious process. Indeed, a different time window must be selected for every frequency of the transfer function estimate that we are interested in. To solve that problem, a tool was developed to create a graphical interface [33, 8] so that the user can select this time window more efficiently.

The tool shows a 3-dimensional time-frequency representation of the signal. In other words, it plots the amount of energy in the signal for each time and each frequency. The center part of Figure 4-3 gives an idea of the picture that is drawn for a noisy output. Since the input signal is known to be a frequency sweep, it appears clearly what part of the signal is actual information and what part is noise. The user can now select a series of points on this picture which defines a polygon on this time-frequency plot. The tool will then only keep the part of the signal inside this polygon. The result of this selection is illustrated on the right picture of Figure 4-3. The left picture represents the input signal.

4.2 Identification

Identification of transfer function through a polynomial fit has attracted a lot of attention [34]. However, when the order of the system gets high, the coefficients

of the optimal polynomial become very sensitive to noise in the transfer function. This is a problem in a flutter boundary determination because we are looking for the correlation between all those coefficients. It was then decided to rely more heavily on a physical model.

4.2.1 Model definition

The model is defined by a state space representation and is parameterized by a finite number of coefficients. This step is very specific to each flutter boundary determination problem and has to be studied with a lot of care each time. It also requires to determine all the known physical properties of the plant and incorporate them into the state spaces matrices which should be dependent on a small number of parameters β_k. For example, in the case described in Chapter 2, the parameters β_k would be some physical values, such as the chord, the moment of inertia or some aerodynamic coefficients. The model should also be dependent on the flight condition through the dynamic pressure q and the Mach number M. In a more synthetic way, this can be expressed by

$$A(\beta_k, q, M). \tag{4.9}$$

The goal of the identification is to estimate those β_k based on data from flight conditions at different q and M.

The major benefit of this method is that it reduces the number of degrees of freedom in the model to the number of β_k that are considered. This means that it forces the results to have a certain structure which makes physical sense. For example, if one coefficient is supposed to increase with say dynamic pressure while an other one should decrease, the proposed methodology does take it into account whereas the damping ratio extrapolation method described in Chapter 2 did not.

4.2.2 Cost definition

The original cost function that was used in the state space identification was a quadratic cost:

$$J = \sum_{i=1}^{N} \text{Trace}(TF(\omega_i) - C(j\omega_i I - A)^{-1}B + D)(TF(\omega_i) - C(j\omega_i I - A)^{-1}B + D)^*, \quad (4.10)$$

where M^* denotes the conjugate transpose of M. The main problem of this cost is that the zeros are not well fitted at all. Indeed the transfer function at a pole has a much higher magnitude than the transfer function at a zero. Therefore, the cost function tends to weight more the frequencies closer to the poles than to the zeros. In the flutter problem, the zeros of the transfer function were also of big importance. Therefore, a logarithmic cost was substituted. For a single input single output system, the cost is

$$J = \sum_{i=1}^{N} (\log |TF(\omega_i)| - \log |C(j\omega_i I - A)^{-1}B + D|)^2. \quad (4.11)$$

The general formulation in the case of a system with m inputs and p outputs is

$$J = \sum_{i=1}^{N} \sum_{k=1}^{m} \sum_{l=1}^{p} (\log |TF_{kl}(\omega_i)| - \log |C_k(j\omega_i I - A)^{-1}B_l + D_{kl}|)^2, \quad (4.12)$$

where TF_{kl} (resp D_{kl}) is the entry of the transfer matrix G (resp. D) located at the k^{th} row and l^{th} column, C_k is the k^{th} row of C and B_l is the l^{th} column of B.

However, this cost only penalizes the magnitude of the transfer function but the phase sometimes did not match the identified model one. The solution was found by penalizing also the phase difference between the estimate and the identified model.

$$J = \sum_{i=1}^{N} \sum_{k=1}^{m} \sum_{l=1}^{p} (\arg(TF_{kl}(\omega_i)) - \arg(C_k(j\omega_i I - A)^{-1}B_l + D_{kl})\|^2 \quad (4.13)$$

The final cost was a logarithmic cost. The additional term on the phase was sometimes added in order to improve the transfer function fit. However, the results when using this additional term did not bring significant improvements in flutter

boundary prediction so no results will be presented with this term.

4.2.3 Estimating the physical parameters

Once the model and the cost function are defined, the last step of the identification procedure is to find among all the admissible state space matrices included in the model, the one that minimizes the cost function

$$J_{opt} = \min_{\beta_k} \sum_i J(\beta_k, q_i, M_i) \tag{4.14}$$

where the subscript i stands for the different flight conditions that are considered.

A wide variety of methods have been proposed in the past to solve optimization problems. They can be decomposed into two distinct categories: the deterministic and the stochastic algorithms. Stochastic algorithms are becoming more and more popular nowadays because of the raise of computer power. The basic stochastic search is called the Monte Carlo search which is basically a pure random search. Modern techniques provided considerable improvement by introducing some strategies in the search. Among those techniques, the most popular ones are the simulated annealing [1] and the genetic algorithm [38, 15]. Simulated annealing replicates a physical process called annealing that heats up a solid until it melts, followed by cooling it down until it crystallizes into a state of perfect lattice. During this process, the free energy of the solid (which corresponds to the cost function to minimize) is minimized but the cooling must be done carefully so that the free energy of the structure is not trapped into a local minimum. Genetic algorithms tend to mimic the evolution of the species, mutating some individuals and eliminating the unadapted ones, leading to the individual that will best fit its environment.

On the other hand, the deterministic methods are usually based on local properties of the function to optimize which will indeed provide only a local minimum. Some global optimization method exist but they can be applied only to functions that have very specific properties. The most commonly used deterministic methods are often based on the first order derivative and they can be improved by second order deriva-

tives. The choice of the algorithm to use in a hard problem for the non expert but some general ideas to optimize the convergence of the algorithm can be found in [10]. The algorithm that was chosen to solve the flutter problem is called a quasi-Newton algorithm. Its principle relies on the first order derivative, calculated analytically and on second order derivative, calculated numerically using the so-called BFGS method. The BFGS method estimates directly the inverse of the Hessian matrix based on the variations of the gradient. A direction of search is defined and a line search is then done. More information about this method can be found in [10, p155]. There is however one task to perform before applying the Newton algorithm: the computation of the gradient. This requires a lot of algebra so it will be presented in Appendix C.

Example

As an example, the state space model described in Chapter 2, Equation (2.17) and (2.18), will be used. The system was simulated with a frequency sweep and it was assumed that only the vertical displacement h was recorded. The procedure presented in this thesis is then applied to this data in order to obtain the air density at which the wing will flutter. It was assumed that 5 parameters of the model were unknown: mass per unit span m, first and second moments of inertia S_α and I_α, bending and torsion frequencies ω_h and ω_α. The goal of this procedure was to recover the values of the original model. The simulations were performed at two different flight conditions (which in this case means two different air densities ρ_1 and ρ_2) and the five unknown parameters were identified using those two experiments simultaneously. Therefore, the cost function had to be augmented:

$$J = J(m, S_\alpha, I_\alpha, \omega_h, \omega_\alpha, \rho_1) + J(m, S_\alpha, I_\alpha, \omega_h, \omega_\alpha, \rho_2). \tag{4.15}$$

Note that in this case, the dynamic pressure dependency was switched to an air density dependency. This does not alter the procedure at all because there is a direct relationship between air density and dynamic pressure. To make the simulation more

realistic, some white noise was added to the output. This white noise had zero mean and a maximum amplitude varying from 0 to 0.5 with a step of 0.1. This highest noise level corresponds to a coherence between the input and the output of a maximum of 0.8 which is approximately what happens for real data. Ten simulations were done for each noise level. Once the state space model was identified, the A matrix of the system was just a function of ρ. The flutter boundary can then be computed. A plot of the expected value of this boundary and its variance for each noise level is shown in figure 4-4. It is shown, that even with no noise, there is some slight error in the identification of those coefficients. This is due to some bias in the transfer function estimation. This problem appear because the system is lightly damped which causes a very high resonance. In those types of system, the damping ratio is much more difficult to identify and this is what causes the slight bias in the transfer function. This problem should not appear as significantly with the real data because there is some additional damping in the system which is not modeled in this example. However, the flutter boundary prediction was very accurate (less than 10% for all the noise levels that were taken). Note also the low level of the variance of the estimate compared to the nominal value. This means that the estimate for the flutter bound is reliable.

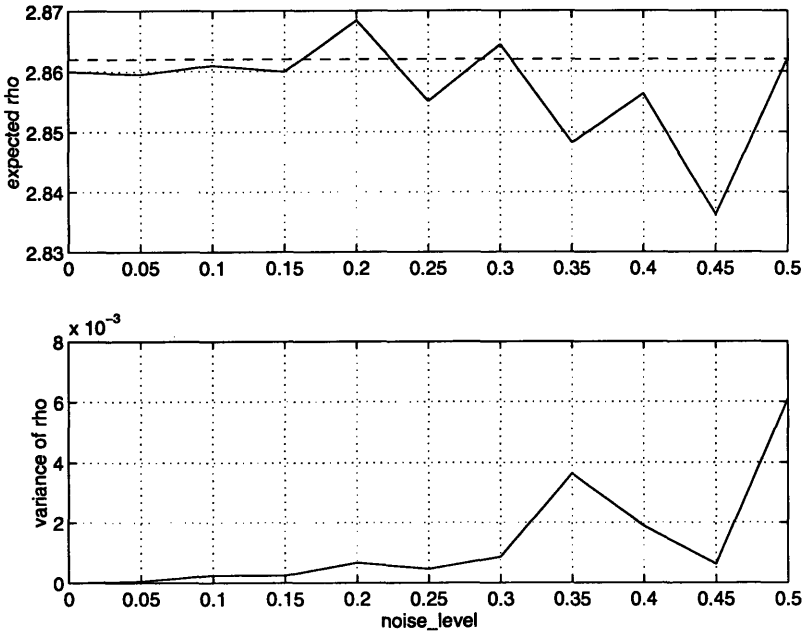


Figure 4-4: This plot shows how the flutter boundary prediction evolves with respect to noise

Chapter 5

Application to wind tunnel data (BACT model)

Work on flutter boundary determination on the Benchmark Active Controls Technology (BACT) model developed at NASA Langley Research Center, was originally done by K. Gondoly and presented in [11]. However, some problems on identification of the experimental data occurred, leading to degraded performance in the flutter boundary determination. Therefore, the identification algorithm presented in Section 4.2 was applied to the BACT model and compared to K. Gondoly's results.

5.1 Presentation of previous research

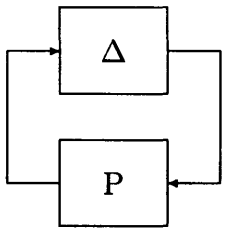
In this section, a review of K. Gondoly's work is presented briefly. The BACT model is a standard NACA 0012 airfoil. It is equipped with a trailing edge actuating surface. The span of the control surface is 30% of the total model span and its chord is 25% of the chord of the wing. Four accelerometers were positioned near the inner leading edge, outer leading edge, inner trailing edge and outer trailing edge and calibrated to provide measurements in g's. An analytical model was then derived using a software package called Interaction of Structures, Aerodynamics and Controls (ISAC) created at NASA Langley Research Center. A detailed description of this package can be found in [14]. This analytical model was parameterized by the Mach number M and

the dynamic pressure q. The dynamics could then be linearized to obtain a state space model of the form

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \left[\begin{bmatrix} A & B \\ C & D \end{bmatrix} + \sum_{i=1}^{2} \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix} \delta_i \right] \begin{bmatrix} x \\ u \end{bmatrix} \tag{5.1}$$

where x represents the state variable, y the outputs and u the inputs. $\delta_{i,i=1,2}$ represent the dynamic pressure q and the Mach number M respectively.

In [11], the author shows how this kind of system can be incorporated into a robustness analysis problem of the form



This is a typical real-μ problem where P represents the nominal plant and Δ the uncertainty block. The latter is a diagonal matrix in which each non zero elements are either q or M. The robustness problem is solved by a maximum singular value determination

$$\frac{1}{\mu} = \min_{\Delta} \{ \overline{\sigma}(\Delta) | \det(I - M(s)\Delta) = 0 \} \tag{5.2}$$

The goal of the work on the BACT model was to predict flutter boundary by using experimental data. This was handled through the following identification procedure that the author developed. The fact is that there exists an infinite number of state space realizations for each system. The main constraint was to obtain all the identified models in the same state basis, which is not guaranteed by any identification procedure. In other words, the states x should always represent the same physical quantity. One way to be sure that the same basis is carried over from each identified

plant is to have as many outputs as there are states and to know the relationship between the outputs and the states. This means that the "C matrix" should be a known full rank square matrix here denoted as $\tilde{C}$. In such a case, the identification can be performed in any state space basis and the following transformation can be applied

$$\begin{cases} T = C^{-1}\tilde{C} \\ \tilde{A} = T^{-1}AT \\ \tilde{B} = T^{-1}B \end{cases} \tag{5.3}$$

where A, B and C are the identified state space matrices and the tilde represents the matrices in the basis of interest.

In the BACT model, there were four outputs and ten states. It was then impossible to use the previous method directly. An intermediary step was introduced, consisting of observing the unmeasured states. The observer was built using the analytical model that was supposed to be close to the true plant. This gave an additional six outputs. Then an identification method called FORSE, developed by R. Jacques for the Space Engineering Research Center (SERC) was used [16]. The state space transformation described in Equation (5.3) was performed, using a $\tilde{C}$ that was also computed analytically.

A series of identifications were realized at different Mach numbers and dynamic pressures. A model of the form of Equation (5.1) was then derived for each experimental condition. The nominal plant was the identified model but the perturbation matrices A_i, B_i, C_i and D_i were all calculated analytically.

5.2 Identification of the BACT model

The conclusion of this experiment was that the major weakness came from the system identification procedure because the unmeasured modes mainly come from the lag terms which are not very much observable. When using experimental data, measurement noise corrupts the observation leading to degraded flutter bounds. Therefore,

the identification method was changed to the one presented in Chapter 4. The experimental data for the BACT model already came into a frequency domain format so it was not necessary to apply the transfer function estimation using a time frequency analysis.

Twelve different experimental data sets at various experimental conditions obtained from simulated data were available for the BACT model and used as a validation process. For each of the twelve experimental conditions taken separately, twelve distinct state space models are identified. The initial guess for the Newton algorithm was always coming from the analytical data. The varying elements of the model (or β_k using the same notations as in Chapter 4) were all the non zero elements of the A matrix. The three other state matrices were all set to their analytical values. Once the twelve state space matrices were identified, a parameterized model of the form of Equation (5.1) was computed using the linear regression for each elements of the matrices A, A_1 and A_2.

Replicating the work of K. Gondoly, the flutter boundary was estimated from each experimental condition by keeping the total pressure constant. This correlated the variation of the dynamic pressure to the Mach number by

$$\delta_M = (\frac{2M + \gamma M^3}{4q})\delta_q, \tag{5.4}$$

reducing the problem to only one degree of freedom q.

Results are presented in Figure 5.1. The top picture shows the results of K. Gondoly and the bottom picture shows the results using the Newton identification method. The results are presented in the same format as in [11]. To be able to compare the performance of the flutter boundary determination from one experimental condition to another, it was necessary to normalize all the results. Since this was analytical data, the flutter boundary was known exactly. The flutter bound estimated with the robustness analysis method can be normalized by the actual bound. This normalized value is plotted on the y-axis of Figure 5.1 and quantifies the accuracy of the prediction. Each flutter bound is estimated from data measure at a certain

nominal dynamic pressure. This value can also be normalized by the actual flutter boundary. This value quantifies how far the nominal value of the dynamic pressure is to the actual flutter one. This quantity is plotted on the x-axis. The shaded region represents what is considered to be an acceptable results. This is somehow arbitrary but it is defined as a triangle because we want more accuracy close to the flutter boundary.

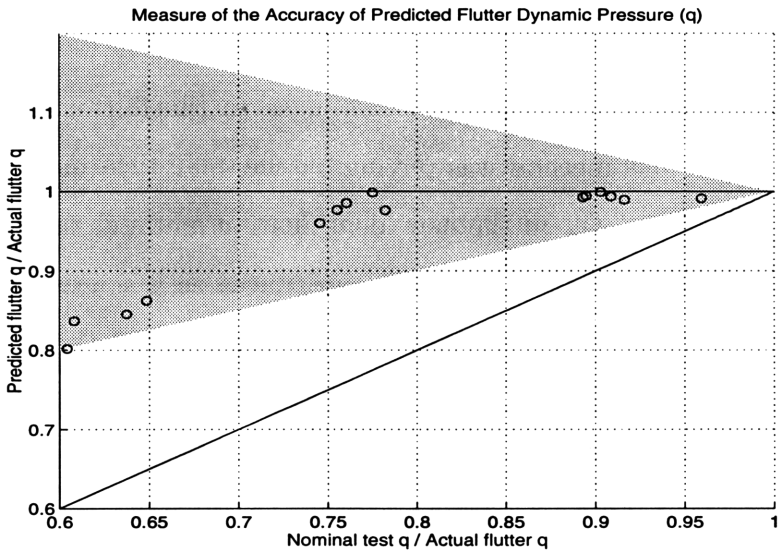
The results using the new identification method are slightly worse than the one presented in [11] when the nominal plant was close to the flutter boundary. The reason is that the previous method was relying on the analytical model to calculate the matrices A_i, B_i, C_i and D_i. In the new identification method, those matrices are estimated from the identified models which contributes as a source of error. Note also that, with the new identification method, the accuracy of the prediction does not depend on how far the nominal point is to the flutter boundary. This results was actually expected because all the bounds were predicted from the same model.

For the experimental data, we are able to present the same type of figure since the true flutter boundary can be determined in a wind tunnel. However, for most of the experimental points, there is no flutter boundary while staying at constant total pressure. The flutter bound was then calculated at constant Mach number as shown in Figure 5.3. In this Figure, the circle represent the nominal points and the cross are the predicted flutter boundary. The curved solid line correspond the the experimental flutter boundary. The diagonal lines are the constant total pressure lines. The numbers on this graph, ranging from 1 to 6 labels the different nominal points. Those labels are carried over on Figure 5.2.

In the method presented in [11], results on experimental data could not be plotted on the same type of graph as Figure 5.1 because the points were out of scale. One reason is that the observer was perfect for the analytical data so the reconstructed states were very accurate. This property was not carried over for the experimental data.

The conclusion of this experiment is that the identification process using a quasi-Newton optimization method is validated. It gave comparable results for the analyti-

Previous results



New results

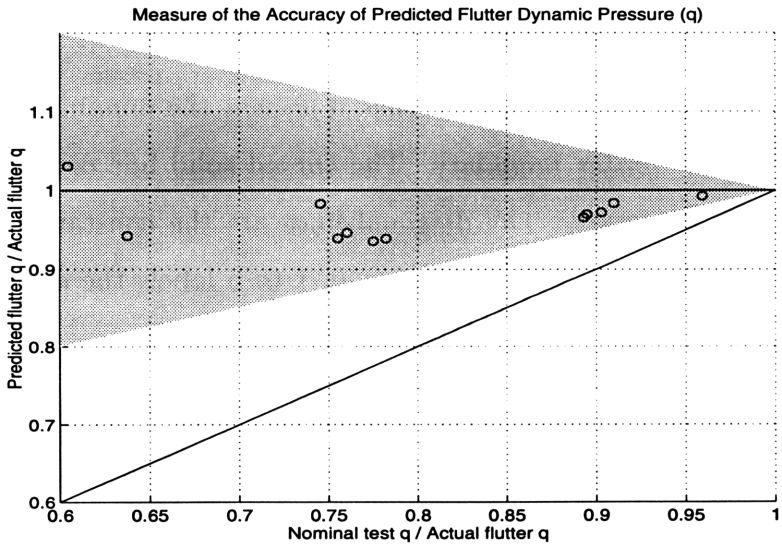


Figure 5-1: Measure of the accuracy of flutter prediction of the BACT model using analytical data.

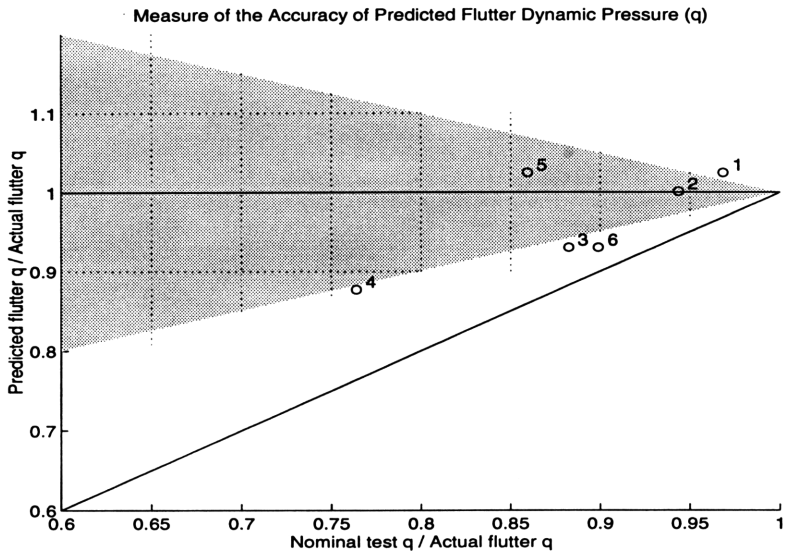


Figure 5-2: Measure of the accuracy of flutter prediction of the BACT model using experimental data.

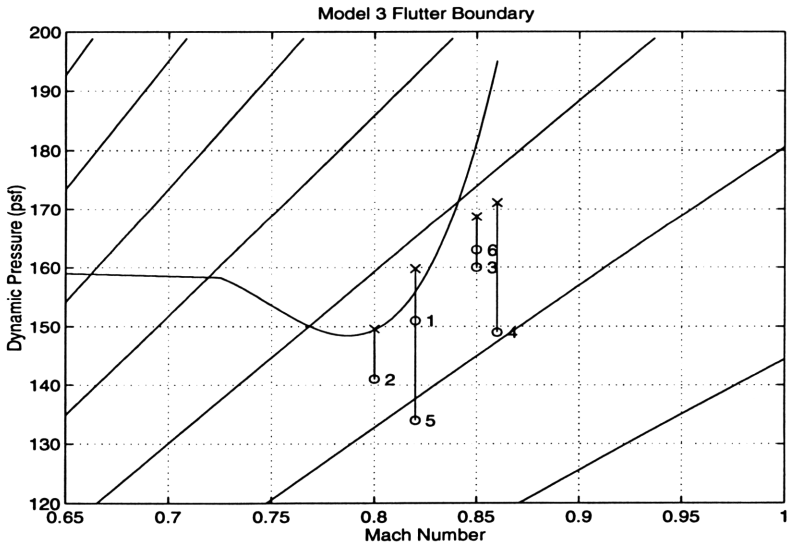


Figure 5-3: Flutter boundary prediction in a Mach vs dynamic pressure diagram

cal model and was able to obtain acceptable results also for experimental wind tunnel data.

Chapter 6

Application to the F18-SRA

The flutter boundary prediction techniques are now being applied to a real aircraft, the F18 System Research Aircraft (SRA). The goal of this project is to provide the flight test engineer with real-time information about how close the pilot is to instability. This would increase the flight envelope of the aircraft considerably, and therefore increase its performance.

6.1 Description of the experiment

The experiment was run at NASA Dryden Flight Research Center, based in Edwards Air Force Base, Calif. This facility possesses a large variety of high technology capabilities including a flight systems laboratory with a diversified capability for avionics system development, a data analysis facility to process flight research data in real time and a high temperature and loads calibration laboratory to ground test structural components. It is well known for supporting the Space Shuttle program and some major current projects include the F-15 ACTIVE, the F-18 High Angle of Attack Research Vehicle (HARV) and the F-18 SRA. The latter aircraft is a modified F-18 which is being flown to test the newest and most advanced technologies such as electrical actuators, fiber optics, flush air data collection systems and flutter clearance techniques.

For flutter boundary prediction, a specific experiment was realized. Its purpose

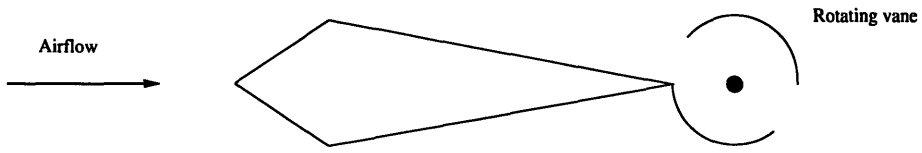


Figure 6-1: Diagram of the DEI exciter

is to excite the structure of the aircraft to obtain a reliable model based on the flight data. This goal was achieved by adding an exciters manufactured by Dynamic Engineering Inc (DEI exciters) at the tip of each wing. This exciter consists of a diamond-shaped, symmetric airfoil section and a rotating slotted cylinder as shown in Figure 6-1. Depending on the position of the slot, the airflow will be directed either above or below the airfoil, therefore creating a positive or negative additional aerodynamic force as shown in Figure 6-2. The magnitude of this force is dependent on the air speed of the aircraft. Its dependence is highly non-linear. A need of measuring this force became very quickly necessary and was obtained by some strain gauges located next to the exciter vanes.

To measure the displacement of the structure, ten accelerometers were placed along the wings and the tail of the aircraft. Figure 6-3 and Table 6.1 details their position.

The inputs were chosen to be linear or logarithmic sinusoidal sweeps spanning the 3 to 30 Hz range. This range was chosen because it is expected to contain all the flutter modes for the F18. The sampling frequency was chosen to be 200 Hz. Each sweep lasted roughly 30 seconds to compromise between the need for reliable information and the requirement to save on operating and maintenance costs.

To be sure to excite the symmetric and the asymmetric dynamics of the aircraft, each test would consist of two sequences. In the first sequence, the two exciters would roughly be in phase, whereas in the second experiment, these exciters would roughly

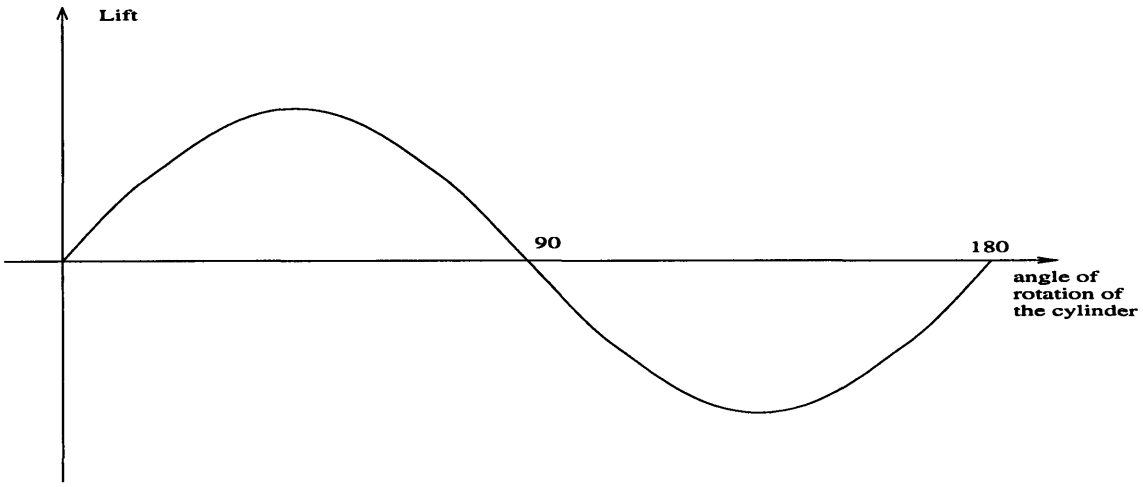


Figure 6-2: Aerodynamic force due to the DEI exciter with respect to its position.

Output	Sensor location
1	left wing forward
2	left wing aft
3	left aileron
4	left vertical tail
5	left horizontal tail
6	right vertical tail
7	right horizontal tail
8	right wing forward
9	right wing aft
10	right aileron

Table 6.1: Position of the accelerometers

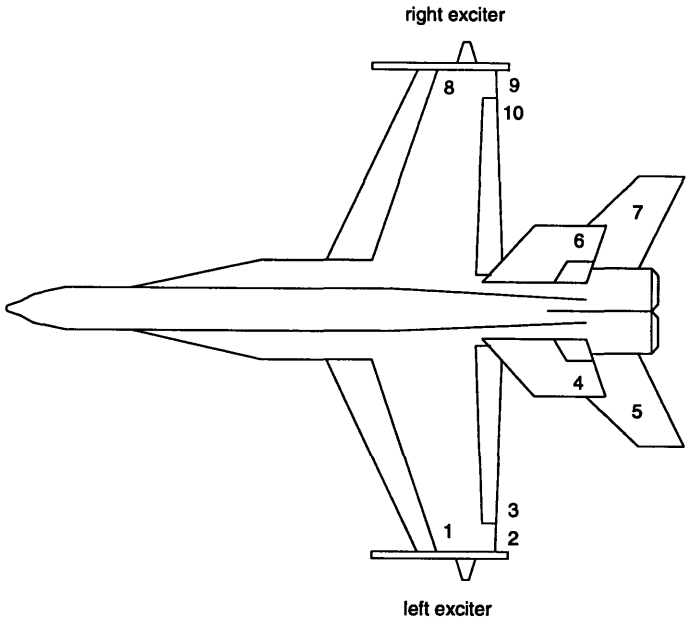


Figure 6-3: Diagram of the F18-SRA with the DEI exciter

be 180 degrees out of phase. The tests were performed at different elevations (10K, 30K or 40K feet) and different Mach numbers (.7, .8, .85, .9 or .95) to obtain a broad range of flight conditions. Some supersonic flight tests were also realized at an elevation of 30,000 feet. The operating conditions at which tests were performed are plotted on Figure 6-4, along with the aircraft flight envelope and the assumed flutter boundary, calculated purely analytically using a p-k iteration method [39].

6.2 Data analysis

The data were collected from real flight tests and were significantly corrupted by atmospheric disturbances. In addition some sensors were suspected to be defective. To investigate both issues, coherence plots between each input and each output were computed. Some corresponding coherence plots may be illustrated in Figure 6-5 which are based on a flight test performed at Mach 0.8 and 10,000 feet.

It may be immediately remarked that on average, the measured coherence is low (no more than 0.8 in most cases). This indicates that the data are contaminated by high levels of noise. From experiment to experiment, the coherence was also found to change significantly (possibly due to different weather conditions). Such a difference in coherence may be used to weight results from different experiments accordingly.

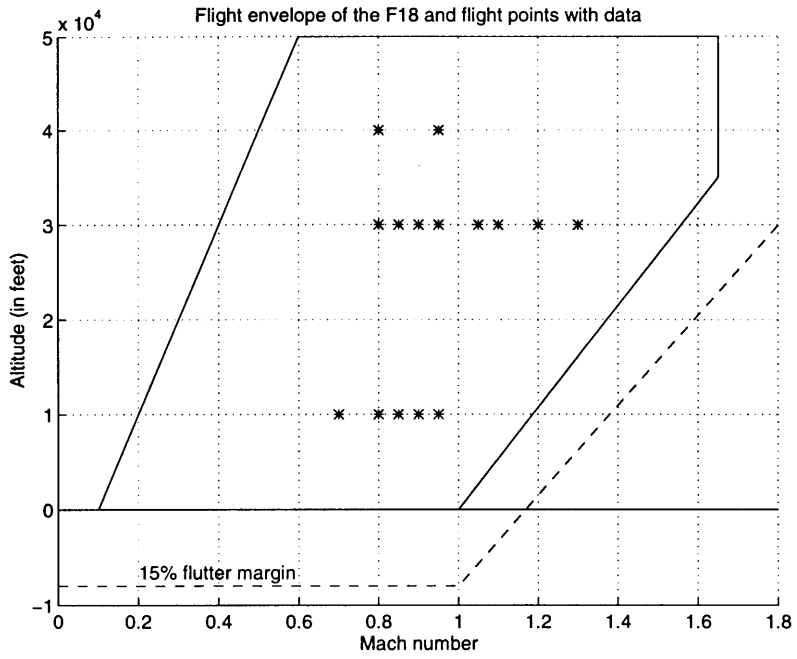


Figure 6-4: Flight envelope of the F18 and flight conditions at which experiments were performed

This issue will not be addressed in this thesis since its purpose is just to present a methodology for flutter determination. To quantify the results that have been obtained, a cost function $\|.\|$ on the coherence plots was defined. It was chosen to be the mean of the coherence between 0 and 50 Hz, bearing in mind that the excitation frequency range is from 3 to 30 Hz. However, due to some non linearities in the exciters, the input signal had some energy up to 50 Hz. Therefore, the coherence can be written as

$$\|\rho\| = \frac{1}{50}\int_0^{50} \rho(\omega)d\omega, \tag{6.1}$$

where ρ represents the coherence function. The resulting scores are presented in Table 6.2 and 6.3. The average of these scores for each output and for both inputs was calculated. It appears that five outputs consistently have a much higher score than any other outputs. These five outputs are the leading and trailing edge accelerometers on each wing, and the right aileron accelerometer. It was decided to discard the latter because it measures also the aileron's own dynamics, which are

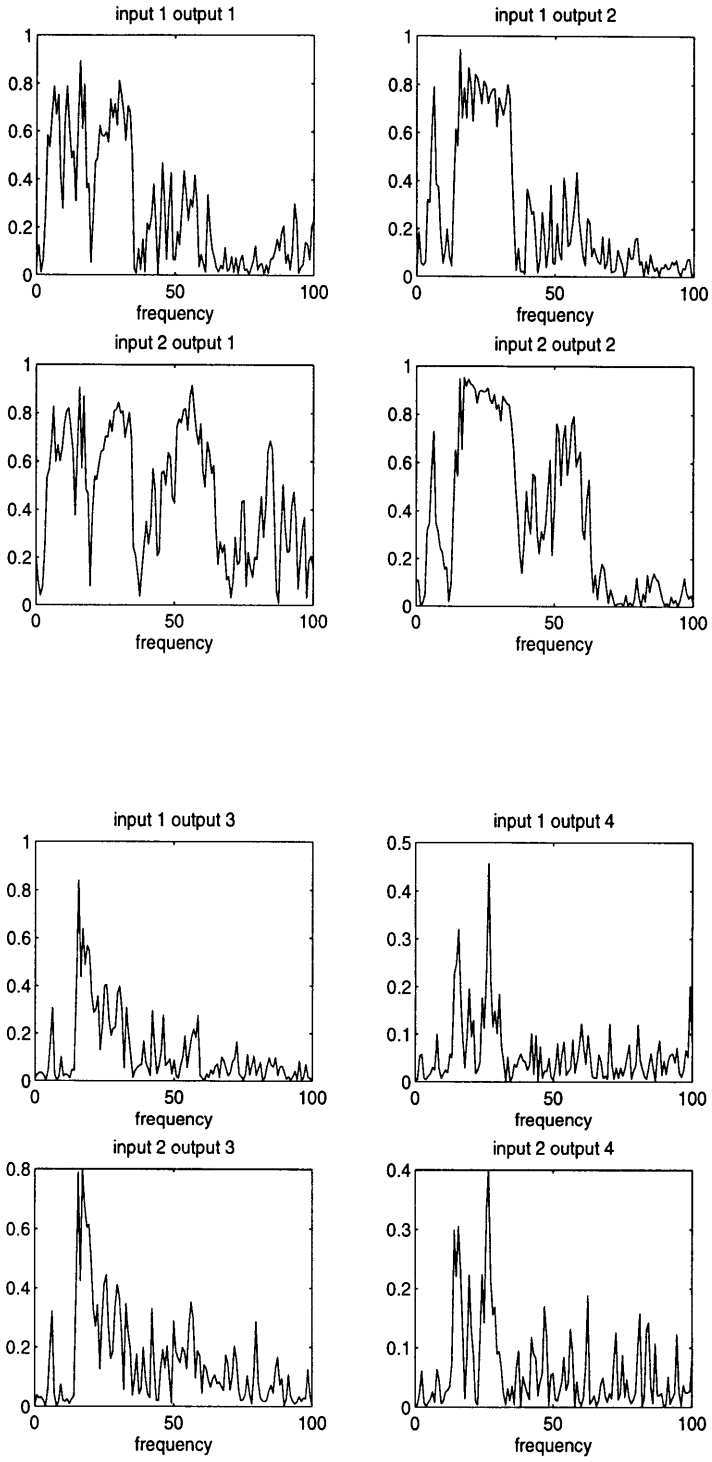


Figure 6-5: Plot of the coherence of the input and output data at Mach 0.8, 10,000 feet

Data	output 1	output 2	output 3	output 4	output 5
1	0.3596	0.339	0.1907	0.111	0.1066
2	0.4153	0.4287	0.1878	0.2063	0.1243
3	0.2896	0.2678	0.1551	0.2104	0.1408
4	0.2417	0.2008	0.1577	0.1600	0.1036
5	0.4639	0.4557	0.3051	0.1470	0.1330
6	0.4266	0.4289	0.2843	0.1877	0.1700
7	0.3992	0.3940	0.2303	0.2100	0.1428
8	0.3470	0.3742	0.2199	0.2655	0.1915
9	0.3943	0.3853	0.2481	0.2042	0.1334
10	0.4121	0.4205	0.2574	0.2590	0.1350
11	0.3661	0.3686	0.2624	0.1018	0.1174
12	0.4416	0.4335	0.3387	0.1731	0.1693
13	0.3799	0.3598	0.2096	0.0922	0.1026
14	0.4934	0.4965	0.2817	0.1952	0.1469
15	0.3605	0.3948	0.2368	0.2359	0.1410
16	0.4007	0.4141	0.2125	0.1585	0.1405
17	0.3292	0.2961	0.2102	0.1438	0.1017
18	0.3799	0.3215	0.1974	0.2762	0.1260
19	0.3866	0.3286	0.2280	0.2701	0.1524
20	0.1479	0.1612	0.3545	0.1998	0.0980
average	**0.3716**	**0.3634**	**0.2341**	**0.1904**	**0.1338**

Table 6.2: Coherence between the right exciter and the five first accelerometers

neglected in the ensuing analysis. When more experience is acquired, this viewpoint may be revised.

Browsing through the public domain literature has led to interesting comparisons. For example, Bucharles, Cassa and Robertier [3] consider the same type of excitation signals for the flutter clearance of the Airbus A320 commercial aircraft. However, the average coherence was ranging from 0.9 to 0.95 which is much higher than the one coming from the F18-SRA. In terms of signal to noise ratio (SNR), this would represent an SNR of 5 to 10, compared to 2.7 for the F18-SRA.

Data	output 6	output 7	output 8	output 9	output 10
1	0.1284	0.1117	0.4754	0.3804	0.3826
2	0.2226	0.1393	0.4680	0.3718	0.3772
3	0.2503	0.1321	0.4103	0.2340	0.2761
4	0.1321	0.1082	0.2867	0.2271	0.2793
5	0.1836	0.1305	0.4967	0.5423	0.4079
6	0.2495	0.1568	0.4445	0.4994	0.3213
7	0.2373	0.1225	0.4702	0.5209	0.3474
8	0.2861	0.1719	0.4085	0.4926	0.2835
9	0.2562	0.1232	0.4737	0.4717	0.3532
10	0.2488	0.1187	0.4437	0.4916	0.3170
11	0.0931	0.1154	0.5352	0.5474	0.5058
12	0.1994	0.1521	0.4563	0.4874	0.4188
13	0.1129	0.1049	0.5006	0.4618	0.4304
14	0.2459	0.1397	0.4773	0.4540	0.3992
15	0.2584	0.1375	0.4048	0.4363	0.3219
16	0.2140	0.1299	0.4230	0.4977	0.3290
17	0.1809	0.0960	0.3550	0.2624	0.3300
18	0.2732	0.1249	0.3153	0.2529	0.3042
19	0.3015	0.1389	0.3564	0.2598	0.4085
20	0.1662	0.1084	0.3721	0.2491	0.4054
average	**0.2120**	**0.1281**	**0.4287**	**0.4070**	**0.3599**

Table 6.3: Coherence between the right exciter and the five last accelerometers

Data	flight number	Mach	Altitude	Input type
1	533	0.85	10k	symm
2	533	0.85	10k	asymm
3	533	0.9	10k	symm
4	533	0.9	10k	asymm
5	531	0.85	30k	symm
6	531	0.85	30k	asymm
7	531	0.9	30k	symm
8	531	0.9	30k	asymm
9	531	0.95	30k	symm
10	531	0.95	30k	asymm
11	532	0.7	10k	symm
12	532	0.7	10k	asymm
13	532	0.8	10k	symm
14	532	0.8	10k	asymm
15	532	0.95	30k	symm
16	532	0.9	30k	symm
17	533	0.95	10k	symm
18	533	0.95	10k	asymm
19	533	0.98	10k	symm
20	533	0.98	10k	asymm

Table 6.4: Flight condition for each data set

6.3 Transfer function estimation

The transfer function of the system was calculated using the method described in Chapter 4.1. It was applied to all the flight test data and Figure 6.6 shows a typical estimate. This specific transfer function was obtained from the flight number 533, at 10,000 feet and Mach 0.8. Only the accelerometers of the right wing are presented in this plot. The inputs are regarded as a symmetric and an asymmetric ones. It is not simple to verify whether the estimate is accurate or not because the real answer is not known. However, some consistency checks may be done between the phase and the magnitude of a transfer function, as well as some consistency among the four outputs that were chosen. In the first transfer function (input 1 to output 1), the magnitude starts with a slope of 40 dB/decade at low frequency and a phase of 180 degrees. The phase then drops to 360 degrees and the magnitude remains constant, at around 40 dB. This would mean that there is a second order pole at 6.5 Hz. It seems that there is also a pole zero cancellation around 13 Hz. However, due to the low resolution of the estimate, this cannot be affirmed by this plot only. The high frequencies (above 18 Hz) are rather noisy, but it seems that the magnitude and the phase are stable, meaning that no poles or zeros are in that region. Looking at the second transfer function (input 1 to output2), it seems that the low frequencies have the same properties as the first plot which is a slope of 40 dB/decade for the magnitude with a phase of 180 degrees. The pole at 6.5 Hz is still detectable on this plot and a pole around 12 Hz is also seen. This confirms the pole zero cancellation of the previous plot. For the asymmetric excitation, two poles could be detected by looking at the transfer function at 8 and 18 Hz. However, it is much harder to correlate the phase in this case, because the poles seem to have a much bigger damping ratio so the variation of the phase with respect to frequency are much smaller.

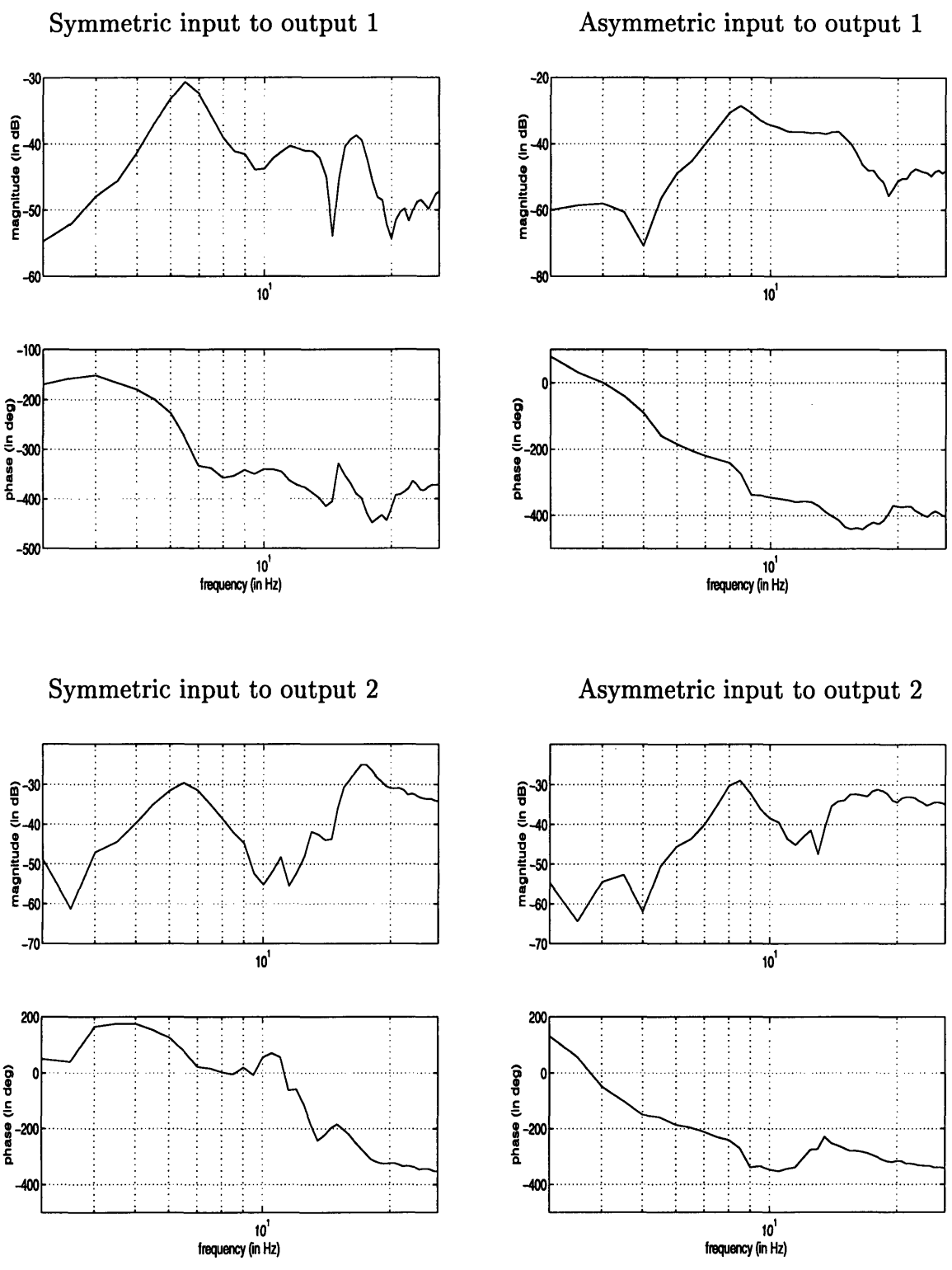


Figure 6-6: Transfer function estimate at Mach 0.8 and elevation of 10,000 feet

6.4 Description of the structural model

For the identification of the state space model, the quasi-Newton algorithm was used. In this method, the first step is to define a structure for the model to be used. A linear time invariant representation was chosen where only the A matrix of the state space model has a dependency in q, the dynamic pressure:

$$\begin{cases} \dot{x} = A(q)x + Bu \\ y = Cx \end{cases} \tag{6.2}$$

More physical constraints need to be imposed to the state matrices in order to capture more accurately the dynamics of the system. Numerous state space model have been proposed in the past but a couple of them seem more popular that others. In one of them [12], the state space matrices are as follow:

$$A = \begin{bmatrix} 0 & I & 0 \\ -M^{-1}K & -M^{-1}C & -M^{-1}P \\ 0 & I & \Delta \end{bmatrix}, \tag{6.3}$$

where I is the identity matrix. The matrices M, K and C are the apparent structural mass, stiffness and damping of the aircraft, which means that the non circulatory part of the aerodynamics is included in the matrices. Those matrices are calculated analytically with a finite element model that was developed at NASA Dryden. Those matrices were used because analytical results for structural dynamics are usually very reliable. The lower right matrix Δ represents the aerodynamic lags. It is a diagonal matrix with real negative eigenvalues. P is a full matrix which appears as the coupling between the lags and the structure. To correlate this part to the Chapter 4, the elements of the P and Δ matrices would be the β_k's, the coefficients to determine.

An other model [29, 18] that is also popular is:

$$A = \begin{bmatrix} 0 & I & 0 \\ -M^{-1}K & -M^{-1}C & -M^{-1}P \\ 0 & Q & \Delta \end{bmatrix}, \tag{6.4}$$

where Q is an other full matrix. In the model described by Equation (6.3), the number of lags is fixed by the number of poles that are taken into account in the structural model of the aircraft. For the model described by Equation (6.4), he number of lags can vary but it usually equals to 2 because additional lags do not improve the flutter boundary prediction significantly (see [29] for reference). The matrices to be determined by the identification are P and Δ, and Q if Equation (6.4) is used.

It was assumed that the DEI exciter only produced a force input on the structure and did not affect the aerodynamics of the wing. This assumption is validated by the size difference between the wing and the exciter. Therefore, the structure of the B matrix is chosen to be

$$B = \begin{bmatrix} 0 \\ \bar{B} \\ 0 \end{bmatrix}$$

where $\bar{B}$ also comes from the finite element model of the aircraft. The sensors are accelerometers so we assumed that they measure only structural displacements and were not affected by the aerodynamics. In the model, this is translated into a matrix C of the form

$$C = \begin{bmatrix} C_1 & C_2 & 0 \end{bmatrix}$$

where C_1 and C_2 are full matrices. Those two matrices C_1 and C_2 are to be determined so they are also part of the parameters β_k to identify.

Those matrices could theoretically also come from analytical results. However, the aircraft was assumed to be symmetric. This assumption is not perfectly valid for a real aircraft since there are always load discrepancies and side slip in a flight test. There is also another source of error which is in the implementation of the sensors.

The positions of the accelerometers are not known precisely enough to rely on the analytical model. Therefore, the C matrix also had to be identified.

6.5 Evaluation of the C matrix

As stated in the description of the model, the C matrix is not considered to be a function of the dynamic pressure. However, discussion with NASA engineers informed us that this matrix was mainly a function of fuel weight. In this thesis, the fuel weight discrepancies will be neglected. This assumption can be validated by the fact that an F18 has some fuel in the wings and in the fuselage but the fuel in the fuselage does not contribute substantially in the flutter boundary. In a flight test, the fuel in the wings is always burnt first. Since the plane has to fly a important amount of time before it reaches the flight test condition, there is very little fuel remaining in the wings when the experiment is run. Thus, the flutter boundary should not be affected very much.

It was then decided to refine the methodology presented in Chapter 4 by calculating the C matrix first. The main benefit is to reduce the amount of parameters β_k in the model. To obtain an accurate value of the C matrix, the identification would have to be done with ground data, so that no aerodynamic lags would corrupt the results. However, such data was not available on the F18-SRA so the flight condition with lowest dynamic pressure was used. The selected condition was at Mach 0.8 and elevation of 40,000 feet where eight different experiments have been performed. For each of these experiments, one C matrix was calculated and Figure 6-7 shows a typical transfer function fit that was obtained by minimizing the logarithmic cost described in Equation (4.11). However, the problem of the initial condition in the Newton algorithm has not been addressed yet. It was solved by using the quadratic cost described in Equation (4.10) whose solution is available through a least square method and requires very little computational time. The cost function was then switched to the logarithmic cost. Due to the fact that the initial guess for the parameters already come from an optimization, the algorithm converged rather quickly to an acceptable

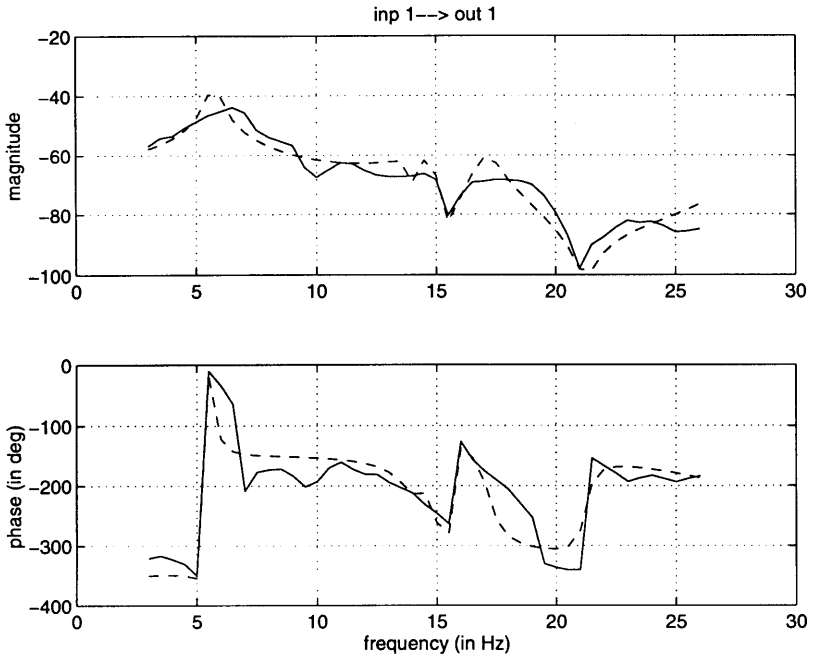


Figure 6-7: Typical transfer function fit

solution.

Eight different C matrices were now available and it was possible to test the sensitivity of the fitting procedure to the data that were used in the procedure. This check was done for each coefficient of the C matrix independently. The mean over the eight experiments as well as the variance were estimated using the following classical statistical formula

$$\begin{aligned} m &= \frac{1}{N}\sum_{i=1}^{N} x_i \\ \sigma^2 &= \frac{1}{N-1}\sum_{i=1}^{N}(x_i - m)^2 \end{aligned} \tag{6.5}$$

where x_i are the coefficients of the C matrix and N is the number of experiments, in this case equal to 8. The normalized standard deviation σ/m of each coefficient with respect to its mean is shown in Figure 6-8. Recall that the normalized standard deviation is the square root of the variance divided by the mean. This standard deviation actually represents the length of the confidence interval of the coefficient by:

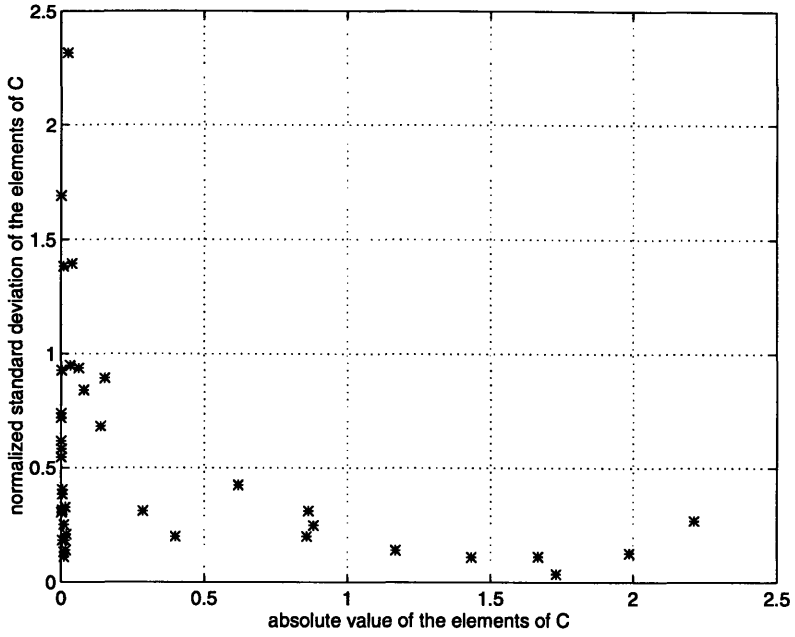


Figure 6-8: Normalized standard deviation vs. mean of the coefficient of C

$$p(x - m < \sigma) \approx 70\% \Leftrightarrow p((x - m)/m < \sigma/m) \approx 70\% \tag{6.6}$$

For elements of the C matrix that are very small, the standard deviation can be very big. This is actually not an important problem because, if those coefficients are small, they should not have a significant influence on the flutter prediction. On the other hand, the coefficients that have a larger value have small normalized standard deviation (below 0.4) which means that the results are consistent from one experiment to an other. Therefore, the mean value for the coefficients of C seemed to be an accurate choice.

6.6 Evaluation of the A matrix

The last step of the identification procedure is to obtain a reliable A matrix parameterized by the dynamic pressure q. Only data from three flight conditions (elevation of 10,000 30,000 and 40,000 feet) were available for each Mach number. However, the flight condition at 40,000 feet has already been used in the identification of the

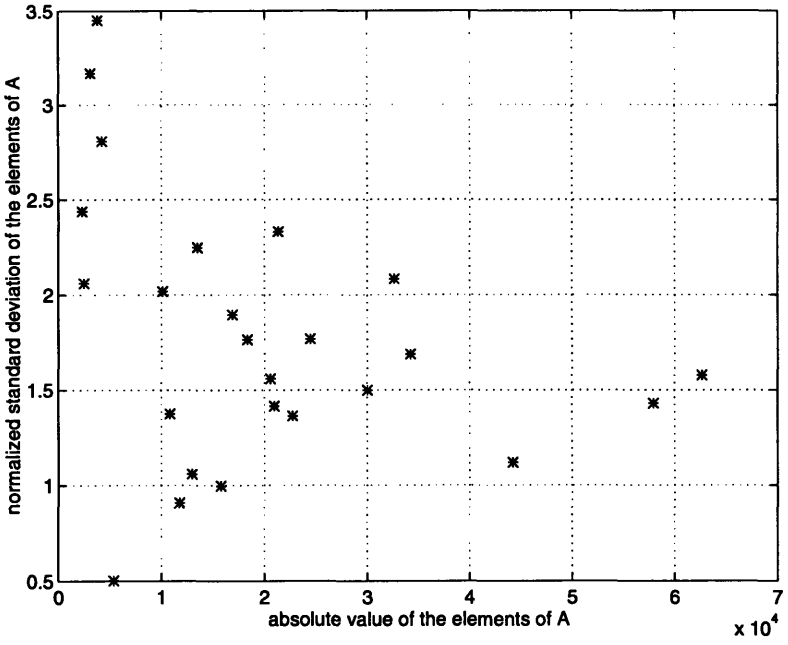


Figure 6-9: Normalized standard deviation vs. mean of the coefficient of A

C matrix so it did not add To solve this problem, the first idea was to identify an A matrix independently for each flight condition and find parameterized model through an interpolation. This did not give good results because there was not enough correlation in the identified coefficients of the A matrix. In order to limit the variability of the lag dynamics, an alternative approach was taken, whereby the coefficients of P, Q and D of Equation (6.4) were assumed to vary linearly with the dynamic pressure, and to be zero at zero dynamic pressure. In addition, the frequency fit procedure was modified such that the fit was performed using simultaneously two data sets coming from two different flight conditions. As for the C matrix case, the normalized standard deviations of the coefficients of the A matrix were plotted with respect to their mean as presented in Figure 6-9. In this case, one can note that the reliability of the identification is much poorer than for the C matrix case.

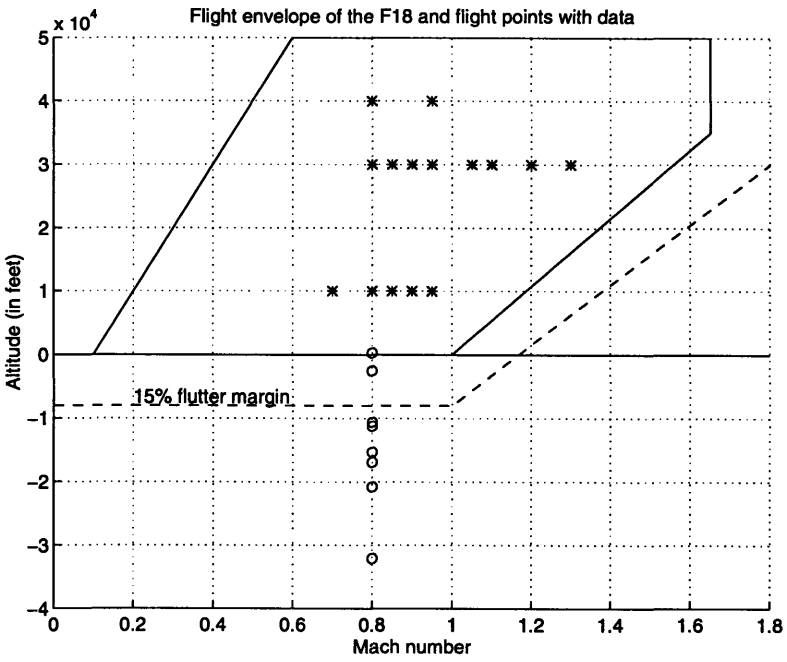


Figure 6-10: Flutter boundary prediction points

6.7 Flutter results

Even though the results on the coefficients of the A matrix were rather poor, it was decided to compute the flutter boundary for all the state space model that were found. Since the system is parameterized by only one value (the dynamic pressure), a line search would give the answer for the flutter boundary. The dynamic pressure can then be converted into an altitude. Figure 6-10 shows the different flutter boundary estimates that were obtained. It is clear that the variance of the estimate is quite big. However, considering that the coefficients in the A matrix are so spread out, it is still interesting to notice that all the points are in the same range of magnitude.

An other important point is that the real flutter boundary is actually unknown, since analytical results do not provide accurate estimations. Thus, none of the provided flutter boundaries can be rejected.

Chapter 7

Conclusion

In this thesis, a methodology to predict flutter boundary has been developed. The procedure consisted of three major steps. First, a transfer function estimation was realized to compress the data into a physically interpretable structure. Then, an identification was performed to obtain a state space model parameterized by the dynamic pressure q. Finally, a stability bound determination gave the flutter boundary.

The transfer function estimation was solved through a time-frequency analysis. The analysis consisted of an appropriate time windowing which optimized the frequency resolution based on the properties of the input signal. This method was specifically developed for systems excited with fast frequency sweeps. This method is the major contribution of the thesis since existing transfer function estimation methods usually assume random excitation. The results of this method were satisfactory for the experimental data. However, in the simulated case, small bias appeared at the poles of lightly damped systems. Even though this error did not lead to large errors in the flutter boundary determination, a more appropriate time windowing can be realized with a wiser use of the graphical interface tool. This would consist of selecting a time window length also based on the *a priori* knowledge of the damping ratio of the poles.

The state space model identification method was based on a quasi-Newton optimization. The cost function chosen in this procedure was based on the transfer function that was previously estimated. The algorithm had very good convergence

properties, meaning that the transfer function was recovered fairly well in every case. This optimization approach worked well on BACT data. However, when applied to the F18-SRA, there was a lack of consistency in the structure of the state space model from one experiment to another. We suspect that this lack of consistency was probably due to the model used since the BACT experiment validated this optimization approach. Another cause for this lack of consistency may also be in the amount of data used for each identification. Only four runs were available for each flight point, and they were used independently in the identification process. This means that each identification used only one run per flight condition.

The stability boundary was determined via a line search. There were no theoretical and practical problems involved with the computation of the bounds. However, there were many of discrepancies from one experiment to another in the estimated flutter boundary in the experimental data case. Such discrepancies were actually expected, given the discrepancies in the identified models of the previous step.

For future work, it would be interesting to develop a more appropriate model of the structure and the aerodynamics of the system that is considered. Developing such a model would lead to more correlation among the elements of the state space matrices. Its main contribution would be to reduce the variance of the identified coefficients β_k.

Another interesting improvement would come from the use of ground data for the identification of the C matrix. In this thesis, the lags in one flight condition (40,000 feet and Mach 0.8) had to be neglected, thus representing a severe approximation. The use of ground data would avoid this approximation because no aerodynamics would corrupt the experimental data. Another benefit of this improvement is that ground tests are much cheaper than flight tests, thus the cost of the overall procedure would be reduced.

A final area of improvement may be in ameliorating the experimental set up. For example, longer sweeps may be useful to obtain high resolution in the transfer function estimation. Furthermore, relocating the sensors positioned on the tail onto the wings may provide better performance in the analysis of the structural dynamics

of the F18. Those sensors were not used here because their coherence plots were too low.

Appendix A

Linearized equation of motion of a typical wing section

In Chapter 2, the equation of motion for a wing subject to aerodynamic forces has been derived. However, the full linear state space model was not presented explicitly. To obtain this model, it is easier to work in the Laplace domain.

Let us recall the equation of motion from Equation 2.17 and 2.18.

$$m\frac{U^2}{b^2}h'' + S\frac{U^2}{b^2}\alpha'' + m{\omega_h}^2 h = -L(\tau) \tag{A.1}$$

$$S\frac{U^2}{b^2}h'' + I_\alpha\frac{U^2}{b^2}\alpha'' + I_\alpha{\omega_\alpha}^2\alpha = M(\tau), \tag{A.2}$$

with the lift and moment presented in Equation 2.15 and 2.16

$$\begin{aligned} L(\tau) &= 2\pi b\rho U^2 \int_{-\infty}^{\tau} \Phi(\tau-\tau_0)[\alpha'(\tau_0) + \frac{1}{b}h''(\tau_0) + (\frac{1}{2} - a_h)\alpha''(\tau_0)]d\tau_0 \\ &\quad +\rho\pi U^2(h'' - a_h b\alpha'') + \rho\pi b U^2\alpha' \end{aligned} \tag{A.3}$$

and

$$M(\tau) = (\frac{1}{2}+a_h)2\pi b^2\rho U^2\int_{-\infty}^{\tau}\Phi(\tau-\tau_0)[\alpha'(\tau_0)+\frac{1}{b}h''(\tau_0)+(\frac{1}{2}-a_h)\alpha''(\tau_0)]d\tau_0$$
$$+a_h b\rho\pi U^2(h''-a_h b\alpha'')-(\frac{1}{2}-a_h)\rho\pi b^2U^2\alpha'-\frac{\rho\pi b^2U^2}{8}\alpha''. \quad (A.4)$$

By taking the Laplace transform of the equation of motion, we obtain

$$m\frac{U^2}{b^2}hs^2+S\frac{U^2}{b^2}\alpha s^2+m{\omega_h}^2h=-L(s) \quad (A.5)$$

$$S\frac{U^2}{b^2}hs^2+I_\alpha\frac{U^2}{b^2}\alpha s^2+I_\alpha{\omega_\alpha}^2\alpha=M(s), \quad (A.6)$$

where s stands for the Laplace variable. The Laplace transform of the lift and the moment are

$$L(s) = 2\pi b\rho U^2\Phi(s)[\alpha s+\frac{1}{b}hs^2+(\frac{1}{2}-a_h)\alpha s^2]$$
$$+\rho\pi U^2(hs^2-a_h b\alpha s^2)+\rho\pi bU^2\alpha s \quad (A.7)$$

and

$$M(s) = (\frac{1}{2}+a_h)2\pi b^2\rho U^2\Phi(s)[\alpha s+\frac{1}{b}hs^2+(\frac{1}{2}-a_h)\alpha s^2]$$
$$+ a_h b\rho\pi U^2(hs^2-a_h b\alpha s^2)-(\frac{1}{2}-a_h)\rho\pi b^2U^2\alpha s-\frac{\rho\pi b^2U^2}{8}\alpha s^2. \quad (A.8)$$

Plugging Equation (A.7) and (A.8) into Equation (A.5) and (A.6) leads to

$$(m\tfrac{U^2}{b^2}+\rho\pi U^2)hs^2+(S\tfrac{U^2}{b^2}-\rho\pi U^2a_h b)\alpha s^2+m{\omega_h}^2h+\rho\pi bU^2\alpha s$$
$$=-2\pi b\rho U^2\Phi(s)[\alpha s+\tfrac{1}{b}hs^2+(\tfrac{1}{2}-a_h)\alpha s^2] \quad (A.9)$$

$$(S\frac{U^2}{b^2} - a_h b\rho\pi U^2)hs^2 + (I_\alpha \frac{U^2}{b^2} + {a_h}^2 b^2 \rho\pi U^2 + \frac{\rho\pi b^2 U^2}{8}\alpha s^2)\alpha s^2 + I_\alpha {\omega_\alpha}^2 \alpha$$
$$+(\tfrac{1}{2} - a_h)\rho\pi b^2 U^2 \alpha s = (\tfrac{1}{2} + a_h)2\pi b^2 \rho U^2 \Phi(s)[\alpha s + \tfrac{1}{b}hs^2 + (\tfrac{1}{2} - a_h)\alpha s^2] \tag{A.10}$$

The Laplace transform of the Wagner equation can be approximated by

$$\Phi(s) = 1 - \frac{0.165}{s + 0.041} - \frac{0.335}{s + 0.32} \tag{A.11}$$

Noting that

$$\frac{s}{s+k} = 1 - \frac{k}{s+k} \tag{A.12}$$

and

$$\frac{s^2}{s+k} = s - k + \frac{k^2}{s+k} \tag{A.13}$$

leads to the three following identities:

$$\Phi(s)\alpha s = \alpha s - 0.165\alpha + 0.165\frac{0.041\alpha}{s + 0.041} - 0.335\alpha + 0.335\frac{0.32\alpha}{s + 0.32}, \tag{A.14}$$

$$\begin{aligned} \Phi(s)\alpha s^2 &= \alpha s^2 - 0.165\alpha s + 0.165 * 0.041\alpha - 0.165\frac{0.041^2\alpha}{s + 0.041} \\ &- 0.335\alpha s + 0.335 * 0.32\alpha - 0.335\frac{0.32^2\alpha}{s + 0.32} \end{aligned} \tag{A.15}$$

and

$$\begin{aligned} \Phi(s)hs^2 &= hs^2 - 0.165hs + 0.165 * 0.041h - 0.165\frac{0.041^2 h}{s + 0.041} \\ &- 0.335hs + 0.335 * 0.32h - 0.335\frac{0.32^2 h}{s + 0.32}. \end{aligned} \tag{A.16}$$

We now introduce two states x_1 and x_2, usually called lags by the following equation:

$$x_1(s) = 0.165 * 0.041 * \frac{(1 - (\frac{1}{2} - a_h)0.041)\alpha - \frac{1}{b}0.041h}{s + 0.041}$$

$$x_2(s) = 0.335 * 0.32 * \frac{(1 - (\frac{1}{2} - a_h)0.32)\alpha - \frac{1}{b}0.32h}{s + 0.32}$$

$$\begin{aligned}
\Phi(s)[\alpha s + \frac{1}{b}hs^2 \;+\; & (\frac{1}{2} - a_h)\alpha s^2] = (\frac{1}{2} - a_h)\alpha s^2 + (1 - (\frac{1}{2} - a_h)0.335)\alpha s \\
+\; & (-0.335 + (0.165 * 0.041 + 0.335 * 0.32)(\frac{1}{2} - a_h))\alpha \\
+\; & \frac{1}{b}hs^2 - \frac{1}{b}(0.165 + 0.335)hs \\
+\; & \frac{1}{b}(0.165 * 0.041 + 0.335 * 0.32)h + x_1 + x_2 \qquad \text{(A.17)}
\end{aligned}$$

By plugging this expression into the equations of motion (A.7) and (A.8), a set of linear equations is obtained.

Appendix B

State space model example

In Chapter 3, a motivational example was presented to show the relevance of a multiple data set algorithm for subspace identification methods. The state space model used in this example is presented here in more detail:

$$\begin{cases} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{cases} \tag{B.1}$$

with

$$A = \left[\begin{array}{cccccccc} 0.89 & -1.5 & -13.1 & -81.9 & -353.5 & -1013.8 & -1957.5 & -1977.6 \\ 0.005 & 1 & 0 & -.2 & -.9 & -2.6 & -5 & -5 \\ 0 & 0.005 & 1 & 0 & 0 & 0 & -0.0084 & -0.0085 \\ 0 & 0 & 0.005 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.005 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.005 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.005 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.005 & 1 \end{array}\right]$$

$$B = \begin{bmatrix} .0047 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Appendix C

Gradient of the cost function used in the identification procedure

The cost function that is considered in the identification procedure is

$$J = \sum_{i=1}^{N} \sum_{k=1}^{m} \sum_{l=1}^{p} (\log |TF_{kl}(\omega_i)| - \log |C_k(j\omega_i I - A)^{-1} B_l + D_{kl}|)^2, \qquad \text{(C.1)}$$

where TF_{kl} (resp D_{kl}) is the entry of the transfer matrix G (resp. D) located at the k^{th} row and l^{th} column, C_k is the k^{th} row of C and B_l is the l^{th} column of B.

Since the Newton search has been selected as the optimization algorithm, the gradient of the cost function has to be calculated. Therefore, the partial derivative with respect to any coefficient of the state space matrix has to be found. Note that the cost function is a sum of terms of the form

$$(\log |TF_{kl}(\omega_i)| - \log |C_k(j\omega_i I - A)^{-1} B_l + D_{kl}|)^2, \qquad \text{(C.2)}$$

so the derivation of the gradient can be restrained to a single input, single output system. Let us start by calculating the partial with respect to a coefficient of the C matrix. An intermediate step is to calculate the differential of J with respect to the row matrix C_k.

$$\begin{aligned} dJ_{C_k} &= \sum_{i=1}^{N} 2(\log(|G_{kl}(\omega_i)|) - \log(|C_k(j\omega_i I - A)^{-1}B_l + D_{kl}|)) \\ & \quad d(\log(|G_{kl}(\omega_i)|) - \log(|C_k(j\omega_i I - A)^{-1}B_l + D_{kl}|)) \end{aligned} \tag{C.3}$$

$$\begin{aligned} dJ_{C_k} &= 2\sum_{i=1}^{N}(\log(|G_{kl}(\omega_i)|) - \log(|C_k(j\omega_i I - A)^{-1}B_l + D_{kl}|)) \\ & \quad \frac{-d(C_k(j\omega_i I - A)^{-1}B_l + D_{kl})}{C_k(j\omega_i I - A)^{-1}B_l + D_{kl}} \end{aligned} \tag{C.4}$$

To obtain the gradient, we need the partial derivative of the cost J with respect to c_{kl}, the element on the l^{th} column of the C_k matrix. In that case, the matrix dC is just a matrix of zeros with a one on the l^{th} column denoted dC_{kl}.

$$\begin{aligned} \frac{\partial J}{\partial c_{kl}} &= \sum_{i=1}^{N} 2\log(|G_{kl}(\omega_i)|) - \log(|C_k(j\omega_i I - A)^{-1}B_l + D_{kl}|) \\ & \quad \frac{-dC_{kl}(j\omega_i I - A)^{-1}B_l}{C_k(j\omega_i I - A)^{-1}B_l + D_{kl}} \end{aligned} \tag{C.5}$$

Let us now calculate the partial derivative with respect to an element of the B matrix. This is done in a similar way as for the C matrix.

$$\begin{aligned} \frac{\partial J}{\partial b_{kl}} &= \sum_{i=1}^{N} 2\log(|G_{kl}(\omega_i)|) - \log(|C_k(j\omega_i I - A)^{-1}B_l + D_{kl}|) \\ & \quad \frac{-C_k(j\omega_i I - A)^{-1}dB_{kl}}{C_k(j\omega_i I - A)^{-1}B_l + D_{kl}} \end{aligned} \tag{C.6}$$

For the differential with respect to A, an intermediate step should be derived which is the value of $d(P^{-1})$.

Let us start by

$$PP^{-1} = I$$

By differentiating both sides of the equality, we get

$$dPP^{-1} + Pd(P^{-1}) = 0.$$

Arranging this equation leads to

$$d(P^{-1}) = -P^{-1}dPP^{-1}. \tag{C.7}$$

We can now apply the same principle as for the B and C matrices to the A matrix:

$$\begin{aligned} \frac{\partial J}{\partial a_{kl}} &= \sum_{i=1}^{N} 2\text{Trace } (\log(|G_{kl}(\omega_i)|) - \log(|C_k(j\omega_i I - A)^{-1}B_l + D_{kl}|) \\ &\quad \frac{-C_k(j\omega_i I - A)^{-1}da_{kl}(j\omega_i I - A)^{-1}B_l}{C_k(j\omega_i I - A)^{-1}B_l + D_{kl}}. \end{aligned} \tag{C.8}$$

Bibliography

[1] E. Aarts and J. Korst. *Simulated Annealing and Boltzmann Machines.* John Wiley & Sons, 1989.

[2] G. Alengrin, R. Bucy, J. Moura, J. Pages, and M. Ribeiro. ARMA identification. *LIDS-P-1588*, 1985.

[3] A. Bucharles, H. Cassan, and J. Roubertier. Advanced parameter identification techniques for near real time flight flutter test analysis. *AIAA-90-1275-CP*, 1986.

[4] Y. M. Cho, G. Xu, and T. Kailath. Fast identification of state space models via exploitation of displacement structures. *IEEE Transaction on Automatic Control*, pages 2094–2017, 1994.

[5] Young Man Cho. *Fast subspace based system identification: theory and practice.* PhD thesis, Stanford University, August 1993.

[6] H. Van der Auweraer, P. Vanherk, P. Sas, and R. Snoeys. Accurate modal analysis measurements with programmed sine wave excitation. *Mechanical Systems and Signal Processing*, pages 301–313, 1987.

[7] L. Duchesne, E. Feron, J.D. Paduano, and M. Brenner. Subspace identification with multiple data sets. In *AIAA Guidance Navigation and Control Conference*, San Diego, CA, July 1996.

[8] E. Feron, A. Turevskiy, L. Duchesne, J. Paduano, and M. Brenner. Improved identification in noisy environments with time-frequency analysis: a flight case. *to be submitted to the AIAA Journal on Navigation, Guidance and Control*, 1997.

[9] Y. Fung. *An Introduction to the Theory of Aeroelasticity.* Dover Publications, 1969.

[10] P. Gill, W. Murray, and M. Wright. *Practical Optimization.* Academic Press Inc., 1981.

[11] K. Gondoly. Application of advanced robustness analysis to experimental flutter. Master's thesis, Massachusetts Institute of Technology, 1995.

[12] K. Gupta, M. Brenner, and L. Voelker. Development of an integrated aeroservoelastic analysis of engineering structures. *NASA RP-1129*, 1991.

[13] H. Hassig. An approximate true damping solution of the flutter equation by determinant iteration. *AIAA Journal of Aircraft*, 8(11), 1971.

[14] S. Hoadley and W. Adams Jr. ISAC v5: Interaction of structures, aerodynamics and controls. Technical report, NASA TM 100666, NASA Langley Research Center, Hampton, VA, April 1994.

[15] J. Holland. *Adaptation in Natural and Artificial Systems.* The University of Michigan Press, 1975.

[16] R. Jacques. *On Line System Identification and Control Design for Flexible Structures.* PhD thesis, Massachusetts Institute of Technology, MAY 1994.

[17] J. N. Juang and R. S. Pappa. An eigensystem realization algorithm for modal parameter identification and model reduction. *Journal of Guidance, Control and Dynamics*, pages 620–627, 1985.

[18] M. Karpel. Design for active flutter suppression and gust alleviation using state-space aeroelastic modeling. *J. of Aircraft*, 19(3):221–227, March 1982.

[19] S. Kung. A new identification and model reduction algorithm via singular value decompositions. *12 th Asilomar conference on circuits, systems and computers*, pages 705–714, 1978.

[20] D.J. Leith, D.J. Murray-Smith, and R. Bradley. Combination of data sets for system identification. In *IEE Poceedings-D*, volume 140, 1993.

[21] R. Lind and M. Brenner. Robust flutter margins of an f/a-18 aircraft from aeroelastic flight data. *submitted to AIAA Journal of Guidance, Control and Dynamics*, 1996.

[22] K. Liu and R. E. Skelton. Q-markov covariance equivalent realization and its application to flexible structure identification. *AIAA Journal of Guidance, Control and Dynamics*, pages 308–319, 1993.

[23] L. Ljung. *Subspace Identification: Theory for the User.* Englewood Cliffs, NJ, 1987.

[24] L. Ljung. *System Identification Toolbox.* The Mathworks Inc., 1995.

[25] J. Makhoul. Linear prediction: A tutorial review. *Proc. IEEE*, 63:561–580, April 1975.

[26] T. McKelvey, H. Akcay, and L. Ljung. Subspace-based multivariable system identification from frequency response data. *IEEE Transaction on Automatic Control*, 41(7):960–978, 1996.

[27] P. Miotto, J. M. Schechun, E. Feron, and J. D. Paduano. High performance bounded control synthesis with application to the f18 harv. In *AIAA Guidance Navigation and Control*, San Diego, CA, July 1996.

[28] B. De Moor, J. Vandewalle, M. Moonen, L. Vandenberghe, and P. Van Mieghem. A geometrical strategy for the identification of state space models of linear multivariable systems with singular value decomposition. *Symposium on Identification and System Parameter Estimation*, pages 700–704, August 1988.

[29] E. Nissim. Reduction of aerodynamic augmented states in active suppression systems. *Journal of Aircraft*, 28(1):82–93, 1990.

[30] A. Oppenheim and R. Schafer. *Discrete Time Signal Processing.* Prentice-Hall, Englewood Cliffs, NJ 07632, 1989.

[31] P. Van Overschee. *Subspace Identification: Theory - Implementation - Applications.* PhD thesis, Katholieke Universiteit Leuven, Belgium, February 1995.

[32] P. Van Overschee and B. De Moor. Subspace algorithms for the stochastic identification problem. *Automatica*, 29(3):649–660, 1993.

[33] X. Paternot. Signal processing via wavelet and identification. Technical report, Swiss Federal Institute of Technology, 1996.

[34] R. Pintelon, P. Guillaume, Y. Rolain, J. Schoukens, and H. Van Hamme. Parametric identification of transfer functions in the frequency domain: a survey. *IEEE Transactions on Automatic Control*, 39(11):2245–2259, November 1994.

[35] O. Rioul and M. Vetterli. Wavelets and signal processing. *IEEE Signal Processing Magazine*, 1991.

[36] W. Rodden and B. Stahl. A strip method for prediction of damping in subsonic wind tunnel and flight flutter tests. *Journal of Aircraft*, pages 7–17, 1967.

[37] R. Scanlan and R. Rosenbaum. *Introduction to the Study of Aircraft Vibration and Flutter.* The Macmillan Company, 1951.

[38] J. Stender. *Parallel Genetic Algorithms: Theory and Applications.* IOS Press, 1993.

[39] L. Voelker. *F-18/SRA Flutter analysis results.* NASA Technical Memorandum, NASA Dryden Flight Reasearch Center, Edwards CA, 1995.

5516-61