METHODS FOR IN-FLIGHT ROBUSTNESS EVALUATION

SUMMARY OF RESEARCH

James D. Paduano *Eric Feron*
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology, Cambridge, MA 02139

Marty Brenner
NASA Dryden Flight Research Center

MS 4840D/RC, Edwards, CA 93523-0273

1 INTRODUCTION

This is a report for the NASA Consortium Program NCC2-5116, entitled 'Methods for In-Flight Robustness Evaluation'. The goal of this program was to combine modern control concepts with new identification techniques to develop a comprehensive package for estimation of 'robust flutter boundaries' based on experimental data. The goal was to use flight data, combined with a fundamental physical understanding of flutter dynamics, to generate a prediction of flutter speed and an estimate of the accuracy of the prediction.

This report is organized as follows: the specific contributions of this project will be listed first. Then, the problem under study will be stated and the general approach will be outlined. Third, the specific system under study (F-18 SRA) will be described and a preliminary data analysis will be performed. Then, the various steps of the flutter boundary determination will be outlined and applied to the F-18 SRA data and others.

2 SPECIFIC CONTRIBUTIONS

The specific contributions of this project include development of a robustness problem formulation for flutter clearance, software development, and testing of general identification methodologies on recorded flight data. The software developments include

• A new subspace identification algorithm that takes into account the presence of multiple data sets,

This research was supported by NASA Ames University Consortium, Re: NCC2-5116.

• A novel algorithm to identify transfer functions in very noisy environments,

• An algorithm to identify a parametric model of flutter dynamics based on frequency response using a Newton-based optimization procedure.

• A procedure to project validated models to a predicted flutter boundary, based on modern robustness analysis methods.

• A first-cut demonstration of the flutter boundary prediction procedure for NASA Langley wind tunnel flutter test data.

In addition to the demonstration on Langley data, all of these algorithms have been tested on flight data provided by Dryden. Their validation is currently under study. The contributions related to subspace identification have all been reported in a paper that appeared at the AIAA Guidance, Navigation and Control Conference [1] and is provided in an Appendix.

3 PROBLEM STATEMENT AND APPROACH

3.1 Problem Statement

Current flutter clearance procedures rely on post-processing of flight test data, which significantly slows the envelope expansion process for new or modified aircraft. *A priori* prediction of the flutter boundary is difficult, and flight test data are currently not used to systematically update these *a priori* predictions. Further, many methods for determining proximity to the flutter boundary are based on either damping ratio identification or tracking the evolution of spectral peaks. Experience and analysis have shown that these measures can be highly nonlinear with respect to flight parameters such as dynamic pressure and Mach number, and thus may be misleading measures of flutter stability (we present a simple illustration of this problem below). A method is needed to alleviate these difficulties, that systematically combines *a priori* knowledge on the aircraft with flight test data into a well-behaved prediction of the flutter boundary, and that does so in near real time so that envelope expansion can proceed more efficiently.

Current practice of using damping ratio as a measure of proximity to instability is motivated by the fact that as an aeroelastic mode approaches neutral stability, its damping ratio approaches zero. Thus one way to predict the flutter boundary is by tracking the damping ratio of each flexible mode as a function of critical parameters (dynamic pressure or Mach number), and subsequently predicting the flutter boundary by

simple linear extrapolation. In many cases however, such a measure can be unreliable. For example, we have plotted in Fig. 1 the evolution of the damping ratio as a function of speed for a typical wing section [8]. We see that a linear extrapolation of the damping ratio will systematically overestimate the flutter boundary. Besides, when far away from the flutter boundary, the damping ratio actually *increases* as the aircraft approaches flutter.

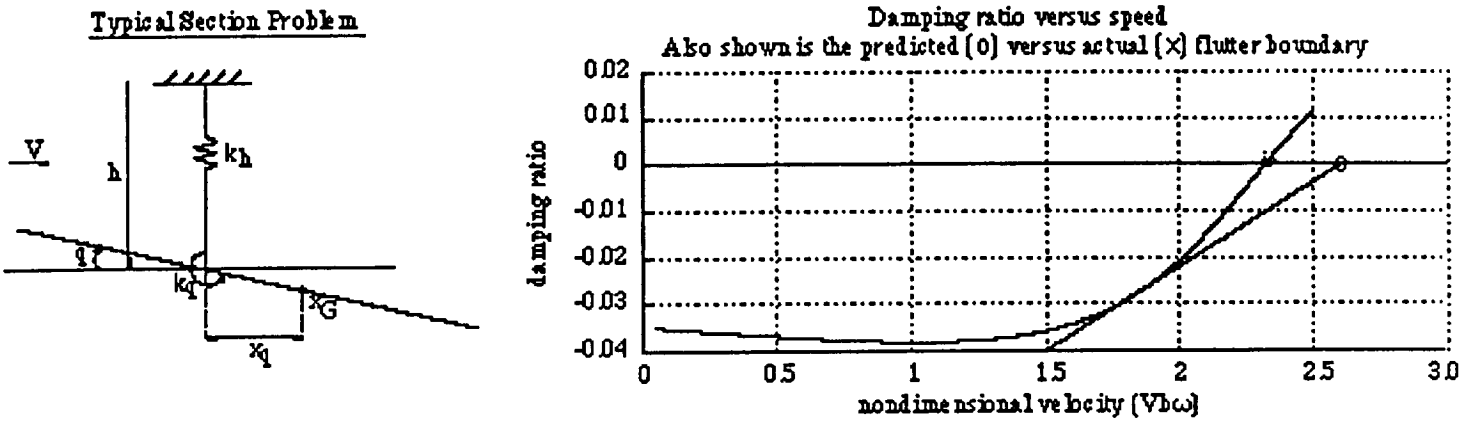
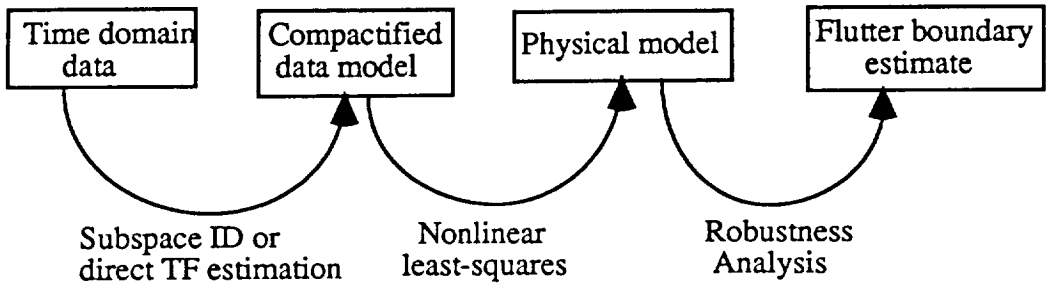


Figure 1 - Evolution of damping ratio vs. speed for a typical section.

3.2 Approach

The approach proposed in this research may be sumarized in the following diagram:



To develop a reliable flutter clearance capability, we cast the flutter clearance problem as a 'robustness problem' typical in control applications. The flutter clearance problem stated as a robustness problem is simply this: based on the currently available

data and models, what is the 'smallest' perturbation to the current flight condition which will drive it to the flutter boundary? Size of the flight condition perturbation is represented as a norm or vectoral distance; this type of measure is typically much more well-behaved than damping ratio or spectral peak amplitude. It is called the *structured singular value* in control theory, and it is the multivariable counterpart of the classical gain margin for single-input, single-output control systems. Several tools are now available that compute it quickly and accurately.

To incorporate flight test data into the robustness problem, we take a unique approach which involves compressing information from multiple data sets, taken at various flight conditions. This is necessitated by the quantity and quality of the data available (large amounts of relatively noisy data), as well as the nonlinearity of the variation of the dynamics with flight condition. Model structure information is retained by the procedure, through a nonlinear least-squares fit of the system dynamics to the acquired data.

To 'compactify' the data and allow more efficient identification, subspace identification methods have been investigated (see for instance De Moor and Vandewalle [1] and Van Overschee and De Moor [2]). Alternatively, we also have investigated the possibility of compressing data by directly generating reliable, de-noised transfer functions. Both technique are essentially parameter-free identification procedures, which provide models that are much reduced in size compared to the original time domain data, but which are completely unstructured and therefore require additional identification to be useful for robustness analysis. This additional identification is performed by adjusting optimally the coefficients in an aeroelastic model that is known *a priori,* such that both models agree as well as possible from an input-output viewpoint. Currently the optimization criterion is a frequency weighted H_2 norm.

4 SYSTEM DESCRIPTION AND PRELIMINARY DATA ANALYSIS

4.1 System description

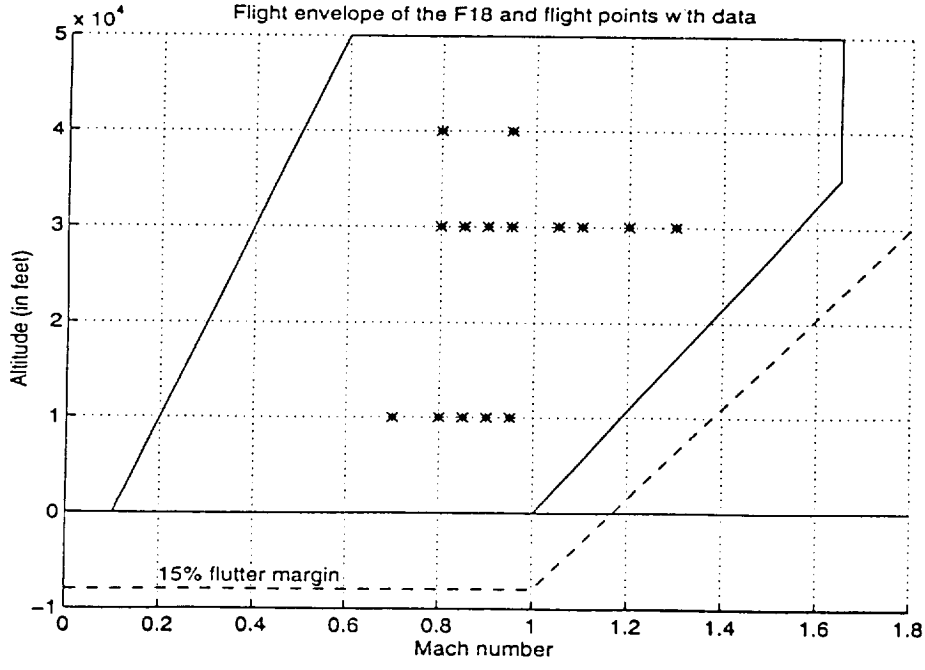
The F-18 SRA (Systems Research Aircraft) is a test aircraft that has been equipped with sophisticated flutter envelope clearance equipment, including a pair of aerodynamic wingtip exciters that provide oscillatory wingtip lift. Ten accelerometers are located on all the aircraft's body to measure the aeroelastic effects of the excitations. The table below summarizes the numbering and position of each sensor.

Output	sensor location
1	left wing forwarc
2	left wing aft
3	left aileron
4	left vertical tail
5	left horizontal tai
6	right vertical tail
7	right horizontal ta
8	right wing forwar
9	right wing aft
10	right aileron

Accelerometer locations

The inputs were chosen to be sinusoidal sweeps spanning the 3-30Hz range. This range was chosen because it is expected to contain all the flutter modes for the F-18. The sampling frequency was chosen to be 200Hz. Each sweep lasted 30 seconds to compromise between the need for reliable information and the requirement to save on fuel and maintenance costs.

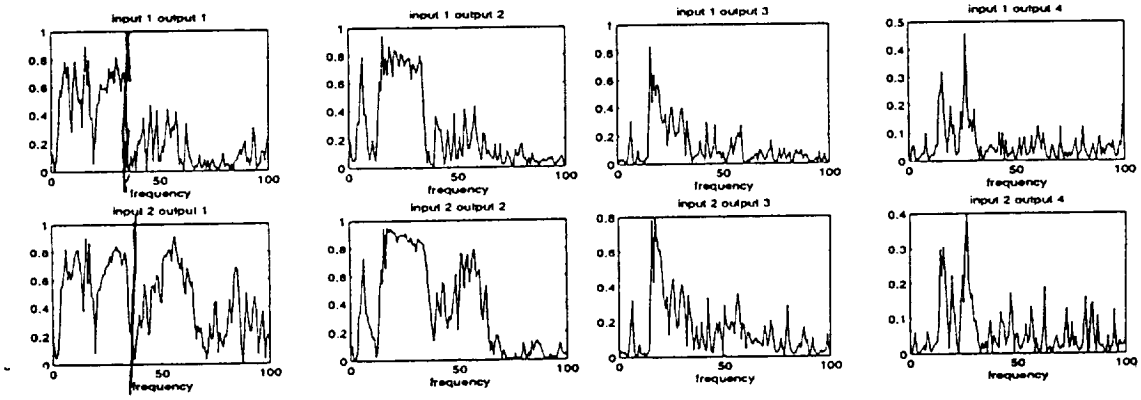
To be sure to excite the symmetric and antisymmetric dynamics of the aircraft, each test consisted of two sequences. In the first sequence, the two exciters were roughly in phase, whereas in the second experiment, these exciters were roughly 180 degrees out of phase. The tests were performed at different elevations (10K, 30K and 40K feet) and different Mach numbers (.8, .85, .9, .95) to obtain a broad range of flight conditions. The operating conditions at which tests were performed is plotted on the graph below, along with the aircraft flight envelope and the assumed flutter boundary.



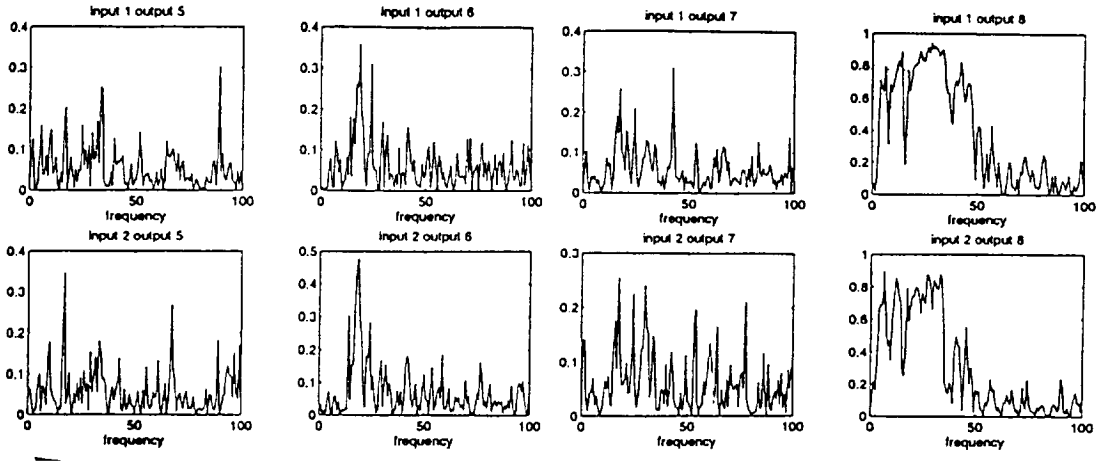
4.2 Data analysis

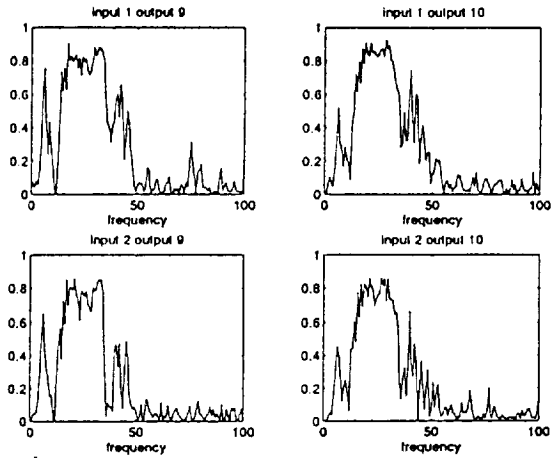
The data were collected from a real flight test and were significantly corrupted by atmospheric and other disturbances. In addition, some sensors were suspected to be defective. To investigate both issues, coherence plots between each input and each output were computed. The corresponding coherence plots may are illustrated in the figure below, which is based on a flight test performed at Mach 0.8 and 10K 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 at hand 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 many experiments differently. To quantify the results that have been obtained, a norm on the coherence plots was chosen to be the mean of the coherence between 0 and 50Hz, bearing in mind that the excitation frequency range is from 3 to 30 Hz. The resulting norms are tabulated below.



COHERENCE PLOTS





COHERENCE PLOTS (CONT'D)

Data	output 1	output 2	output 3	output 4	output 5	output 6	output 7	output 8	output 9	output 10
1	0.3596	0.339	0.1907	0.111	0.1066	0.1284	0.1117	0.4754	0.3804	0.3826
2	0.4153	0.4287	0.1878	0.2063	0.1243	0.2226	0.1393	0.468	0.3718	0.3772
3	0.2896	0.2678	0.1551	0.2104	0.1408	0.2503	0.132	0.4103	0.234	0.2761
4	0.2417	0.2008	0.1577	0.16	0.1036	0.1321	0.1082	0.2867	0.2271	0.2793
5	0.4639	0.4557	0.3051	0.147	0.133	0.1836	0.1305	0.4967	0.5423	0.4079
6	0.4266	0.4289	0.2843	0.1877	0.17	0.2495	0.1568	0.4445	0.4994	0.3213
7	0.3992	0.394	0.2303	0.21	0.1428	0.2373	0.1225	0.4702	0.5209	0.3474
8	0.347	0.3742	0.2199	0.2655	0.1915	0.286	0.1719	0.4085	0.4926	0.2835
9	0.3943	0.3853	0.248	0.2042	0.1334	0.2562	0.1232	0.4737	0.4717	0.3532
10	0.412	0.4205	0.2574	0.259	0.135	0.2488	0.1187	0.4437	0.4916	0.317
11	0.366	0.3686	0.2624	0.1018	0.1174	0.093	0.1154	0.5352	0.5474	0.5058
12	0.4416	0.4335	0.3387	0.1731	0.1693	0.1994	0.152	0.4563	0.4874	0.4188
13	0.3799	0.3598	0.2096	0.0922	0.1026	0.1129	0.1049	0.5006	0.4618	0.4304
14	0.4934	0.4965	0.2817	0.1952	0.1469	0.2459	0.1397	0.4773	0.454	0.3992
15	0.3605	0.3948	0.2368	0.2359	0.141	0.2584	0.1375	0.4048	0.4363	0.3219
16	0.4007	0.4141	0.2125	0.1585	0.1405	0.214	0.1299	0.423	0.4977	0.329
17	0.3292	0.2961	0.2102	0.1438	0.1017	0.1809	0.096	0.355	0.2624	0.33
18	0.3799	0.3215	0.1974	0.2762	0.126	0.2732	0.1249	0.3153	0.2529	0.3042
19	0.3866	0.3286	0.228	0.2701	0.1524	0.3015	0.1389	0.3564	0.2598	0.4085
20	0.1479	0.1612	0.3545	0.1998	0.098	0.1662	0.1084	0.372	0.2491	0.4054
Average	0.3716	0.3634	0.2341	0.1904	0.1338	0.212	0.1281	0.4287	0.407	0.3599

Coherence with right input

Data	output 1	output 2	output 3	output 4	output 5	output 6	output 7	output 8	output 9	output 10
1	0.5206	0.4965	0.2216	0.1156	0.1017	0.1426	0.1065	0.3824	0.3183	0.3162
2	0.5028	0.4804	0.2159	0.2094	0.127	0.223	0.1346	0.4342	0.364	0.3646
3	0.364	0.324	0.1797	0.2195	0.1426	0.2454	0.1332	0.3578	0.2279	0.2545
4	0.5093	0.4043	0.2668	0.212	0.14	0.2515	0.1285	0.206	0.181	0.189
5	0.6025	0.6028	0.3812	0.1508	0.1319	0.1932	0.1339	0.4525	0.4953	0.378
6	0.6403	0.6283	0.382	0.214	0.1683	0.2544	0.1659	0.3896	0.4309	0.296
7	0.564	0.5566	0.2841	0.217	0.1501	0.2622	0.1281	0.4025	0.4741	0.3419
8	0.5479	0.5365	0.2857	0.2653	0.189	0.294	0.1793	0.3593	0.4184	0.2704
9	0.5542	0.5324	0.2878	0.2292	0.1294	0.2937	0.1328	0.4029	0.4403	0.3499
10	0.5515	0.57	0.2985	0.2947	0.138	0.3049	0.1283	0.4229	0.4387	0.3168
11	0.6424	0.6497	0.4346	0.1046	0.1153	0.11	0.1187	0.3403	0.3364	0.3307
12	0.6624	0.6957	0.5274	0.1653	0.1855	0.2265	0.1349	0.2974	0.3284	0.2718
13	0.5671	0.5621	0.2719	0.0938	0.0979	0.1118	0.1059	0.3939	0.3687	0.3474
14	0.5366	0.54	0.2942	0.1952	0.1469	0.2357	0.1316	0.4585	0.4358	0.3915
15	0.5434	0.5416	0.2877	0.2359	0.1419	0.2887	0.1368	0.3587	0.3907	0.3253
16	0.5391	0.5511	0.2817	0.1585	0.1397	0.2398	0.1349	0.3722	0.434	0.3061
17	0.4597	0.353	0.2596	0.1495	0.102	0.1842	0.0934	0.3133	0.2536	0.3184
18	0.4812	0.3562	0.228	0.275	0.1347	0.2726	0.131	0.2849	0.2485	0.3037
19	0.4825	0.3584	0.3545	0.2696	0.1634	0.2795	0.1299	0.2931	0.2374	0.3343
20	0.2177	0.1808	0.1448	0.1742	0.1167	0.1542	0.1468	0.1948	0.1835	0.1998
Average	0.5245	0.496	0.2944	0.1974	0.1381	0.2284	0.1317	0.3559	0.3503	0.3103

Coherence with left input

Data	flight#	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

Flight condition for each data set

The average of this norm for each outputs 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 accelerators on each wing, and the right aileron accelerator. It was decided to discard the latter, because it also measures the aileron's

own dynamics, which are 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 [7] use signals with average coherences ranging from 0.9 to 0.95 to perform the flutter analysis of Airbus commercial aircraft.

5 DATA COMPRESSION

Two data compression approaches have been investigated during this research. First, subspace identification algorithms have been tested and improved. Second, because the data under consideration are so noisy, an alternative direct transfer function evaluation scheme was developed, which is robust to a large range of noise disturbances.

5.1 Subspace identification: review of existing techniques and new developments.

As most of the information gathered during this phase of the project has been already reported in the conference paper [1], this section will only highlight its main findings. The attached appendix contains the conference paper.

Subspace identification techniques represent a way to perform system identification in an unstructured way, using state-of-the-art results in matrix computations, including least-squares and total least-squares techniques. Subspace identification techniques are essentially "black-box" algorithms, which attempt to infer a low-order, state-space model that "best" explains a given set of inputs and outputs. To evaluate how useful these procedures are for aerospace applications such as flutter boundary determination, the F-18 SRA data were used as benchmark data. Many subspace algorithms currently available were tested on these data, especially the algorithms by Cho and also by DeMoor [1]and VanOverschee [2].

The results of this study led to the following conclusions:

(i) The collected data are often distributed over many flights at the same flight condition. This is incompatible with existing subspace identification software, which were all built for single I/O data sequences. To remedy this problem, the basic principles of subspace identification were reviewed and adapted to handle many, uncorrelated data sets. This technique has since then found applications in other areas of interest to NASA-Dryden,

including the reliable identification of the F-18 SRA lateral-directional dynamics for nonlinear simulation calibration and autopilot design purposes.

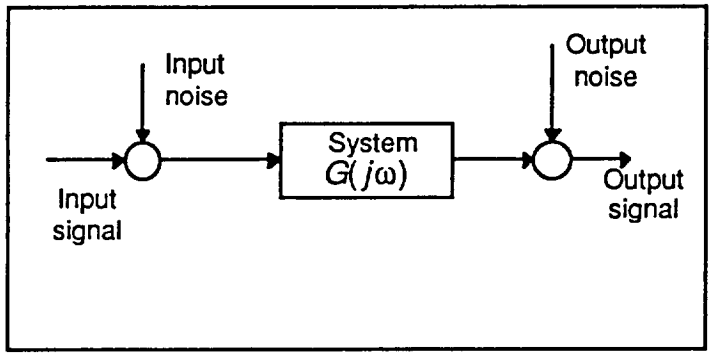
(ii) The data are very noisy and subspace identification methods are on the whole unable to cope with these levels of noise. Besides, different subspace identification techniques do not all share the same performance. As a result, an alternate data compression scheme whereby transfer functions are directly estimated has been devised.

5.2 Direct transfer function estimation

This method is based on the a priori knowledge that the input signal is a frequency sweep. Thus it is possible to predict *a priori* at what time the relevant frequencies should appear in the input and output signals.

5.2.1 Single input-single output case

We first consider the single input-single output case. The assumptions for this method is that the excitation is a frequency sweep. Process and observation noises are present as shown in the following figure.



The step by step procedure of the transfer function estimation is described as follow.

Step 1: localization of the relevant frequencies by filtering the signal with a set of narrow band pass filter centered around the frequencies ω_i. Filtering the input and the output does not alter the identification. Indeed if the input and the output of the system are filtered with the same linear time invariant filter F, the filtered output appears as the output of the system excited by the filtered input.

Step 2: localization of the relevant time information by windowing each filtered signals. The chosen window has to be the same for the input and the output and is chosen to be centered at the point where the filtered input has the maximum amplitude. At this step, the two signals should appear as sine functions.

Step 3: calculation of the estimate at the frequency ω_i by the ratio of the output Fourier transform of the windowed signal to the input one:

$$\hat{G}(j\omega_i) = \frac{\sum y_{mes}(n) e^{-2\pi j\omega_i n}}{\sum u_{mes}(n) e^{-2\pi j\omega_i n}}$$, where y_{mes} and u_{mes} denotes the measured signals.

Let us now look at the properties of this estimate. Let n_1 and n_2 be the input and output noises, respectively. Then we have

$$\hat{G}(j\omega_i) = \frac{G(j\omega_i)(Y(j\omega_i) + N_1(j\omega_i)) + N_2(j\omega_i)}{U(j\omega_i)}$$

$$= G(j\omega_i) + G(j\omega_i)\frac{N_1(j\omega_i)}{U(j\omega_i)} + \frac{N_2(j\omega_i)}{U(j\omega_i)}$$

If the two noises are assumed to be unbiased, then $\overline{N}_i(j\omega_i) = \sum n_i(k) e^{-2\pi j\omega_i k} = 0$. Therefore, $\overline{\hat{G}}(j\omega_i) = G(j\omega_i)$, i.e. the estimate is unbiased.

The variance of the noise is given by $\sigma_N^2 = E(\sum n(k)^2) = N_0\sigma^2$, where σ^2 is the variance of the noise, and N_0 is the number of points in the time window. The variance of the transfer function estimate can now be calculated, assuming that the two noises are uncorrelated:

$$\sigma_G^2 = \left|\frac{G(j\omega_i)}{U(j\omega_i)}\right|^2 N_0\sigma_1^2 + \left|\frac{1}{U(j\omega_i)}\right|^2 N_0\sigma_2^2$$

Assuming that the windowed input is a sine function, $U(j\omega_i) = \frac{N_0}{2}U$, with U being the amplitude of the input, the variance of the estimate is

$$\sigma_G^2 = |G(j\omega_i)|^2 \frac{4\sigma_1^2}{N_0U^2} + \frac{4\sigma_2^2}{N_0U^2}$$

5.2.2 Multiple input single output case

At this point, let us assumed that the system has m inputs. A set of m independent experiments is necessary to use the proposed method and all the excitations are assumed to be frequency sweeps. The first step is identical to the single input case. The property

that the filtered output will be identical to the output associated with the filtered inputs is still satisfied since, in the Fourier domain, the matrix of the filter is a scalar matrix which commutes with any other matrix. For the second step, the procedure is the same but one of the m inputs has to be taken as a reference in order to select the time window. For the third step, the Fourier transform of every signal has to be calculated and the estimate is given by

$$\begin{bmatrix} \hat{G}_1(j\omega_i) \\ \vdots \\ \hat{G}_m(j\omega_i) \end{bmatrix} = \begin{bmatrix} U_{1_1}(j\omega_i) & \cdots & U_{1_m}(j\omega_i) \\ \vdots & & \vdots \\ U_{m_1}(j\omega_i) & \cdots & U_{m_m}(j\omega_i) \end{bmatrix}^{-1} \begin{bmatrix} Y_1(j\omega_i) \\ \vdots \\ Y_m(j\omega_i) \end{bmatrix},$$

with U_{k_l} the l^{th} input of the k^{th} experiment. Note that the input matrix has to be full rank in order to apply this formula (if it is not full rank, we conclude that t here not enough information in the data to estimate the transfer function). With the same assumption for the noise, the estimate still has no bias. The variance of the estimate is given by

$$\sigma_G^2 = E\left(\begin{bmatrix} \hat{G}_1(j\omega_i) - G_1(j\omega_i) \\ \vdots \\ \hat{G}_m(j\omega_i) - G_m(j\omega_i) \end{bmatrix} \begin{bmatrix} \hat{G}_1^*(j\omega_i) - G_1^*(j\omega_i) & \cdots & \hat{G}_m^*(j\omega_i) - G_1(j\omega_m)^* \end{bmatrix} \right),$$

where E stands for the expected value. By plugging in the value of the estimate, this becomes

$$\sigma_G^2 = \begin{bmatrix} U_{1_1}(j\omega_i) & \cdots & U_{1_m}(j\omega_i) \\ \vdots & & \vdots \\ U_{m_1}(j\omega_i) & \cdots & U_{m_m}(j\omega_i) \end{bmatrix}^{-1} \begin{bmatrix} \sum_p G_p(j\omega_i) N_{inp_{1_p}}(j\omega_i) + N_{out_{1_p}} \\ \vdots \\ \sum_p G_p(j\omega_i) N_{inp_{m_p}}(j\omega_i) + N_{out_{m_p}} \end{bmatrix} \cdots$$

$$\begin{bmatrix} \sum_p G_p(j\omega_i) N_{inp_{1_p}}(j\omega_i) + N_{out_{1_p}} \\ \vdots \\ \sum_p G_p(j\omega_i) N_{inp_{m_p}}(j\omega_i) + N_{out_{m_p}} \end{bmatrix}^* \begin{bmatrix} U_{1_1}(j\omega_i) & \cdots & U_{1_m}(j\omega_i) \\ \vdots & & \vdots \\ U_{m_1}(j\omega_i) & \cdots & U_{m_m}(j\omega_i) \end{bmatrix}^{-*}$$

Since all the noises are assumed independent, this equation becomes

$$\sigma_G^2 = \begin{bmatrix} U_{1_1}(j\omega_i) & \dots & U_{1_m}(j\omega_i) \\ \vdots & & \vdots \\ U_{m_1}(j\omega_i) & \dots & U_{m_m}(j\omega_i) \end{bmatrix}^{-1} \dots$$

$$\begin{bmatrix} \sum_p \left|G_p(j\omega_i)\right|^2 N_{inp_{1_p}}{}^2 + N_{out_{1_p}}{}^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sum_p \left|G_p(j\omega_i)\right|^2 N_{inp_{m_p}}{}^2 + N_{out_{m_p}}{}^2 \end{bmatrix} \dots$$

$$\begin{bmatrix} U_{1_1}(j\omega_i) & \dots & U_{1_m}(j\omega_i) \\ \vdots & & \vdots \\ U_{m_1}(j\omega_i) & \dots & U_{m_m}(j\omega_i) \end{bmatrix}^{-*}$$

Since the middle matrix is a scalar matrix (diagonal with all its diagonal terms equal), it commutes with the input matrix. Thus the variance is proportional to the singular values of the inverse of the input matrix. This matrix is independent of the noise. Its singular values give some information on how well the inputs have been chosen for identification of the system.

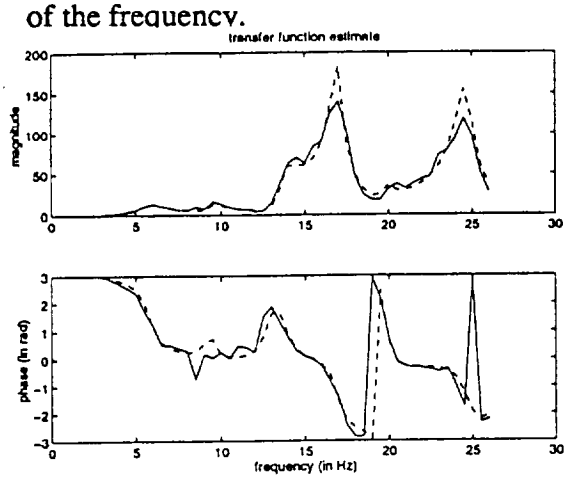
5.2.3 Numerical experiment

This method was applied to simulation data using a model of the structural dynamics of the F18. The system has two inputs. The two sets of experiments needed for the estimation were symmetric and asymmetric excitation, with equal amplitude *U* for each input. For each time *k*, the process noise *n(k)* was chosen to be a random variable uniformly distributed over the interval [*-U/4 U/4*]; thus the variance of each of the inputs is $U^2/(4*12)$. No sensor noise was added in this experiment. The time window length was chosen to be 128 points. The variance was estimated with the simulated data by using the classical estimator for the variance, $1/(N-1)\sum x^2$, where N is the number of experiments and x is the error of the estimate. One hundred simulations were made in order to obtain a 90% confidence interval for the variance of approximately 5%. The input matrix can be written as follow: $N_0/2\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$. Since the output noise is 0, the theoretical variance of the estimate is

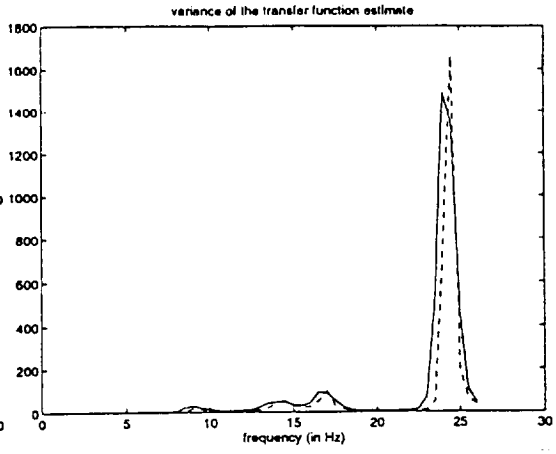
$$\sigma_G^2 = 8/N_0^2U^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} G_1(j\omega_i) + G_2(j\omega_i) & 0 \\ 0 & G_1(j\omega_i) + G_2(j\omega_i) \end{bmatrix} N_0U^2/(4*12)$$

$$= 1/3N_0 \begin{bmatrix} G_1(j\omega_i) + G_2(j\omega_i) & 0 \\ 0 & G_1(j\omega_i) + G_2(j\omega_i) \end{bmatrix}$$

Note the variance of the transfer function estimate is independent of the noise amplitude. The variance was compared with the theoretical results and plotted below. In this case, the standard deviation of the variance estimate is 5% of the estimate and is independent of the frequency.



Solid line - estimated transfer function
Dashed line - actual transfer function

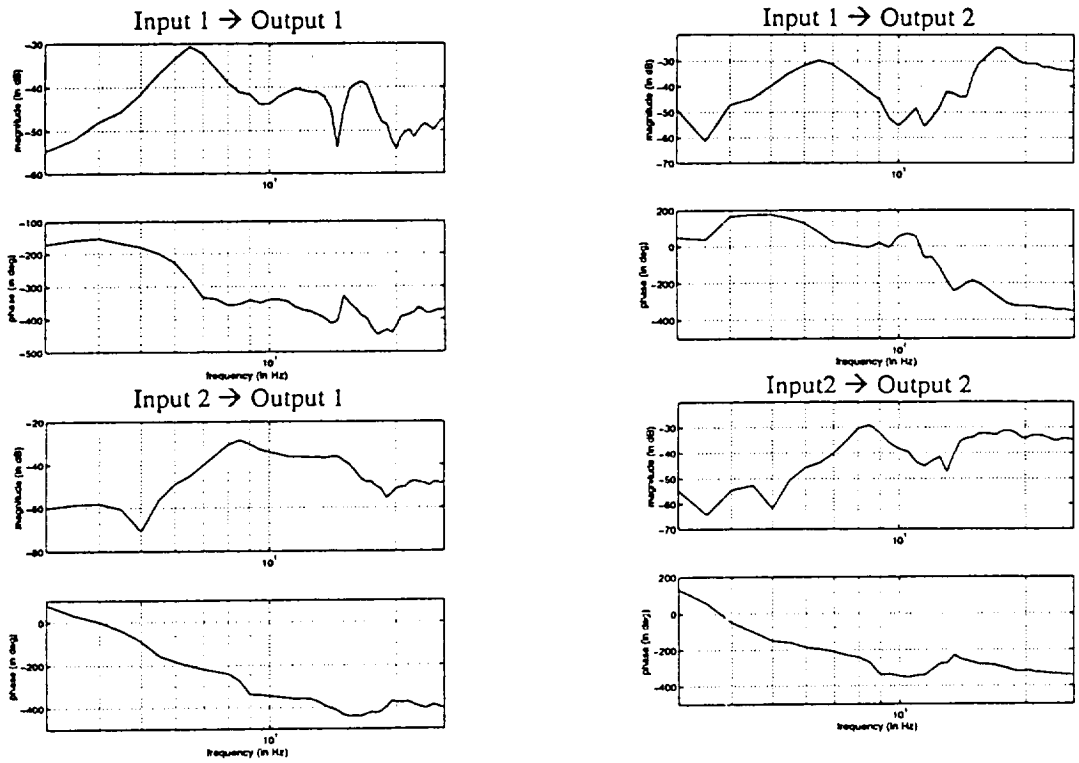


Solid line - theoretical variance
Dashed line - estimated variance

5.2.4 Application to F18-SRA data

This method was then applied on the real flight data using only the four outputs chosen in Section 2. The four transfer functions plotted below were obtained from F-18 SRA flight data taken at 10,000 feet and Mach 0.8. The first transfer function (input 1 to output 1) 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 is constant, at around 40 dB. This means 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 no pole or zeros are in this region. Looking at the second transfer function (input 1 to output 2), 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 (which is the same as -180). 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 magnitude of the transfer function at 8 and

18 Hz. However, it is much harder to correlate the phase in this case, because its variation with frequency is much smoother.



6 IDENTIFICATION OF STATE-SPACE DYNAMICS BASED ON COMPRESSED DATA

Once the data are compactified (either in state-space via subspace identification or via direct transfer function estimation), it is necessary to fit a physical model of the system to them,

where the input/output form of this model is given by

$$\begin{cases} \dot{X} = AX + BU \\ Y = CX + DU \end{cases}$$

The parameters in the fitting procedure are the significant elements of the system matrices (A, B, C, and D), where the system states are the physical states, and the initial guess and structure of the model (structure is represented by the zero and identity

elements of the system matrices) come from *a priori* modeling. In its basic version, the fitting operation consists of minimizing the H_2 norm of the difference between the system defined by the matrices (A, B, C, and D) over the relevant coefficients (such as stability derivatives and aerodynamic lags) and the experimental, compressed data. In a more sophisticated version, a series of fits is done first over a range of flight conditions. These fits are then interpolated using a specific parameterization with respect to, for example, Mach number and dynamic pressure q.

6.1 Nonlinear fit via Newton algorithm

To find a system of the form (1) that best matches the experimental data, we propose to constraint some elements of the matrices to be constant in accordance to the nominal flutter model that was chosen. The chosen cost function is the H2 norm of the difference between the estimated transfer function and the transfer function of the identified model. Even though some elements of the matrices were fixed, the optimization problem is still unconstrained in the mathematical sense since all the coefficients that are allowed to vary are totally free.

The H2 norm for a system is given by:

$$J = \int_0^{+\infty} Tr((C(j\omega - A)^{-1}B + D - G(j\omega))(C(j\omega - A)^{-1}B + D - G(j\omega))^*)d\omega$$

The gradient of J with respect to all the matrices needs to be evaluated. With respect to the matrix A, the gradient is:

$$\frac{\partial J}{\partial A} = 2\,\mathrm{Re}\left(\int_0^{+\infty} Tr((C(j\omega - A)^{-1}dA(j\omega - A)^{-1}B)(C(j\omega - A)^{-1}B + D - G(j\omega))^*)d\omega\right)$$

By using the multiplication commutation inside the trace operator, we have

$$\frac{\partial J}{\partial A} = 2\,\mathrm{Re}\left(\int_0^{+\infty} Tr(((j\omega - A)^{-1}B)(C(j\omega - A)^{-1}B + D - G(j\omega))^* C(j\omega - A)^{-1}dA)d\omega\right)$$

With the same procedure, the gradient with respect to B, C and D can be evaluated:

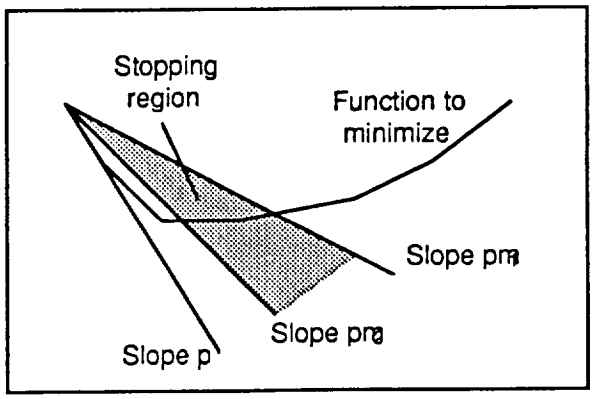
$$\frac{\partial J}{\partial B} = 2\,\mathrm{Re}\left(\int_0^{+\infty} Tr(C(j\omega - A)^{-1}B + D - G(j\omega))^* C(j\omega - A)^{-1}dB)\,d\omega\right),$$

$$\frac{\partial J}{\partial C} = 2\,\mathrm{Re}\left(\int_0^{+\infty} Tr((j\omega - A)^{-1}B(C(j\omega - A)^{-1}B + D - G(j\omega))^* dC)d\omega\right), \text{ and}$$

$$\frac{\partial J}{\partial D} = 2\,\mathrm{Re}\left(\int_0^{+\infty} Tr((C(j\omega - A)^{-1}B + D - G(j\omega))^* dD)\,d\omega\right)$$

These integrals are well approximated by a discretization (in frequency) at all the points at which the transfer function has been estimated. The gradient of the function J can then be derived from the above formulas.

For a Newton optimization, it is also necessary to calculated the Hessian H of the function. However, this operation is computationally very intensive; it is generally more efficient to use a so-called quasi-Newton method instead. This method estimates H based on the variation of the value of the function and of the gradient. A efficient method to estimate H is the BFGS method. Once H is estimated, the direction of search is calculated by $d = H^{-1}\nabla J$. An optimization of J along this search direction is then performed to find the minimum J in the current search direction (i.e. the step length is optimized). The classical 1D Newton method can be used at this stage, but it is then necessary to calculate the derivative in the direction of search at each step. Studies on optimization have shown that it is not necessary to find a very accurate minimum in the current search direction because, at the next iteration, the direction of search is going to change. The little gain obtained by optimizing in each search direction can usually be avoided if it is time consuming. The procedure that was used was developed by de Wolf. The derivative in the current search direction is estimated only at the starting point and noted p. Then a stopping region is defined by two coefficients m_1 and m_2 (these can be tuned but can be set to .4 and .6 respectively as a first try). m_1 and m_2 define two straight lines whose slope are respectively pm_1 and pm_2. The region between the two lines is the 'stopping region'. A binary search is then performed until a point of the surface J falls inside the stopping region.



The whole process is then iterated until a convergence is reached. The main drawback of this method is that it guaranties convergence to a local minimum only, and there is absolutely no way to distinguish, even *a posteriori*, if the point that was reached is a global or local optimum.

6.2 Application to F-18 SRA data

6.2.1 Physical flutter model

The first step to use the quasi Newton algorithm is to determine which elements in the state space model are fixed and which are to be optimized. Even though numerous flutter models have been proposed, a literature search showed that the most common is written as follows:

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

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 that the non-circulatory part of the aerodynamics is included in the matrices. The lower-right matrix Δ represents the aerodynamic lags and is a diagonal matrix with real negative eigenvalues. P is the coupling term between the lags and the structure.

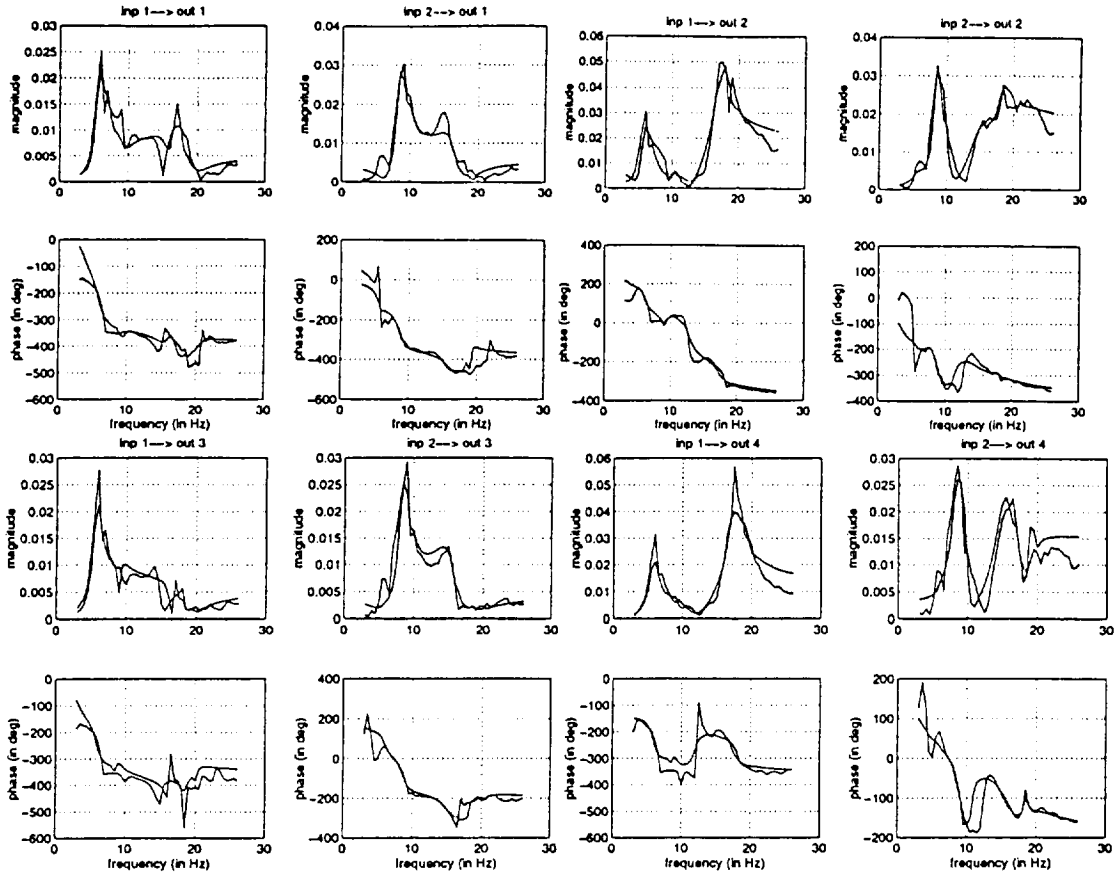
6.2.2 Actual data and high-order model

A finite element study of the structural dynamics of the F18 was computed by NASA Dryden. The full model was composed of 14 symmetric modes and 14 anti symmetric modes. 13 of them were in the 3-20Hz range and 11 of them may be involved in the flutter mechanism. It was therefore decided to reduce the model to 11 modes (or 22 states). The first experiment that was tried was to put all the uncertainty into the lags. The matrices M, K and C were set by reducing the theoretical model to keep only the 22 states of interest. For this, the state space system was diagonalized and only the modes with the appropriate eigenvalues were kept reduced. The matrices P and Δ could vary in the optimization and D was set to 0. It was assumed that the actuator acted as a force input on the structure and did not affect the aerodynamics around the wing. Therefore, the structure of the matrix B is chosen to be $B=\begin{bmatrix} 0 \\ \bar{B} \\ 0 \end{bmatrix}$. The sensors are assumed to measure only structural displacements and no aerodynamics at all. In the model this shows in a matrix C of the form: $C=\begin{bmatrix} C_1 & C_2 & 0 \end{bmatrix}$. Finally, it was found to be necessary to add an additional constraint in the Newton algorithm that forced the lag coefficients to stay above a certain limit (so that the identified model remains stable).

The optimization set up as described above saturates the constraints on the lags, and does not converge to even a local minimum. Thus it was decided to optimize over a single output, 2 outputs, and 3 outputs, to determine the effect of trying to fit more and more outputs on the optimization process. First, only one output was used and the algorithm ran with exactly the same structure of matrices. A stable local minimum which did not saturate the constraints was found for this case. A measure of the accuracy of the identification was defined to be the ratio between the value of the cost function J and the nominal H2 norm of the estimated transfer function, evaluated in %. This measure went down to 13.3% for the single-output test. When a second output was added, the accuracy of the identified model degraded to 20.6%. When a third output was added, the accuracy of the identified model stalled at 31.1%. Thus we conclude that it is progressively more difficult to explain all of the data with the single model that we are using; when all four outputs are chosen the search fails altogether.

At this point, the order of the system to be identified was further reduced to 6 modes. Two things motivated this approach. First, computation time grows exponentially with both the number of parameters to fit *and* the order of the system, since it is necessary to calculate the inverse of a complex matrix of the size of A at every iteration. The second motivation was to detect whether some of the modes in the models were absolutely necessary. In this test, the system was modeled with only 6 modes. This approach converged with all four outputs, and yielded a system of order 18. The solution found had an accuracy of 33%. The figure on the following page shows the frequency domain plots of all 8 transfer functions, together with the results of the optimization. Despite the large value of the optimality measure, the basic shapes of the transfer functions have been captured consistently across all eight transfer functions.

Based on these tests, we conclude that by reducing the model order, the difficulties encountered with convergence were alleviated to a sufficient degree that the four-output test was able to converge. The trend of error in the fit continues as expected, however: the percent error is highest for the four-output identification. It is important to note, however, that the reduction in the model order did not significantly effect the overall accuracy of the fit (31% vs. 33%). Thus the hypothesis that some of the structural modes are indeed unnecessary to explain the data appears to be a valid one. Follow-on work is focusing on judicious model order reduction, as well as optimization of the mode shape parameters (contained in the C matrix) *separately* from the lag parameters.



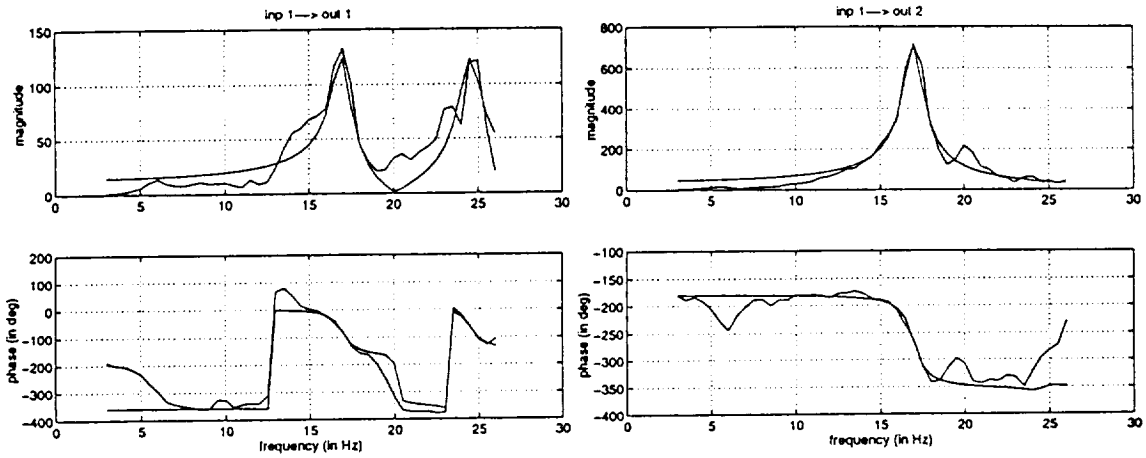
6.2.3 Simulated data and low-order model

Based on the results of the previous section, a test was executed using simulated data and a very low order system. As before this was motivated by the desire to reduce computation; for a very low order representation the number of local optimum should also be reduced. Therefore, since the Newton algorithm converges to any local minimum, the chances that the global minimum is reached are increasing. At this point, the physical model that was used primarily was totally ignored. The number of modes was set to two because two resonance appeared clearly on the transfer function estimates. The initial guess for the A matrix was of the following form:

$$A=\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k_1 & 0 & -c_1 & c_{12} \\ 0 & -k_2 & c_{12} & -c_2 \end{bmatrix}$$

k_1 and k_2 were set so that their square root is equal to the approximate value that was read on the transfer function plot. The values of c were set so that the system is stable. These values were also chosen rather small since it is well known that structural dynamics

eigenvalues are lightly damped. The number of outputs was reduced to two because the symmetric and anti-symmetric part were assumed to be completely decoupled. The convergence of the Newton algorithm was quite fast as expected. More importantly, the fits were acceptable (less than 10% of relative error, see comparisons below), even though the model used for simulation had 14 modes and the model used for identification had only two modes. The experiment was tested for three different altitudes (10K, 30K and 40K feet) and the Mach number was kept constant at .8. At every flight point, both the symmetric and anti symmetric cases were tested. The results of the identification were always good and the convergence fast if a reasonable initial guess was utilized.



At this point, even if we assume that the optimum that was found was the global minimum, there is no guaranty that the three models are in the same basis. Therefore, some additional constraints were added to the second and third identification, forcing them to have the same C matrix as in the first identification.

Another experiment was tried with the same set of data and the same parameterized model which consist in identifying the three Taylor expansion matrices in one single Newton algorithm. Since only three flight points were considered, the Taylor expansion should exactly match the previous results at the three elevations that were considered. The initial guess for this experiment was taken as the result of the previous Newton optimization for the nominal point, and the two other matrices of the Taylor expansion were set to 0. This means that the initial guess assume a constant A matrix with respect to the parameter.

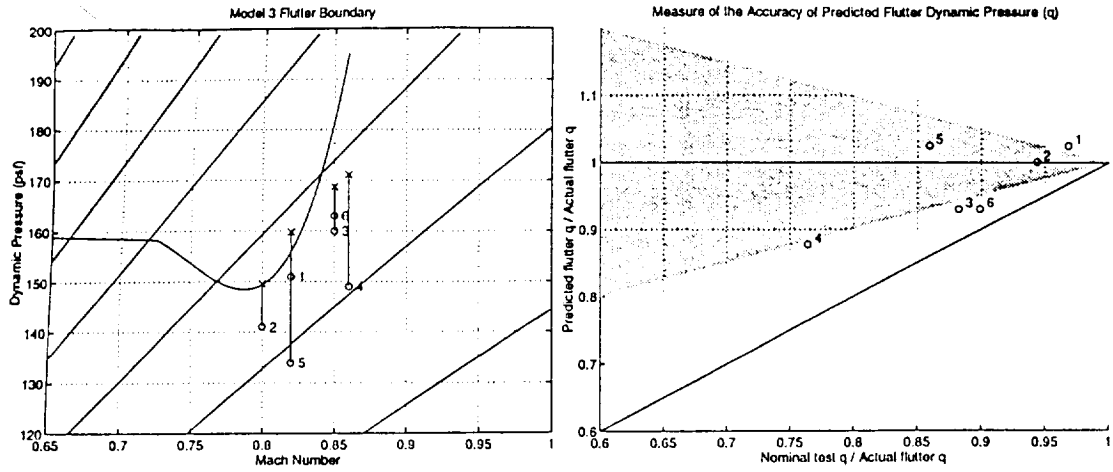
6.2.4 Conclusions about F-18 SRA model identification

Several critical issues were identified for the success of applying the Newton algorithm to identification of the structural dynamics of the F18. The computation time involved to solve the problem can be quite high if the model order and number of free parameters is not carefully considered. Choosing a reliable model order is a critical step, and further work is needed to determine what constraints should be applied and how knowledge of the physical system can be incorporated. Finally, the coefficients of the matrices should not be completely independent; at the very least they should be tied to dynamic pressure in a physically meaningful way - this is the focus of current research.

6.3 Analysis of the Benchmark Active Control Technology (BACT) Demonstrator at NASA Langley

Initial development of procedures for flutter boundary prediction were also tested in cooperation with NASA Langley, utilizing data from the BACT rig. This rig is a 32" wing section mounted in Langley's transonic wind tunnel facility [3]. The purpose of the device is to test flutter control strategies in a benchmark environment. Because of this role, high frequency forcing can be introduced via trailing edge flaps and upper and lower surface spoilers. Spectral transfer function estimates of the aeroelastic behavior have been obtained, and detailed modeling has been conducted. Again, a nominal, finite element model of this system was available to be fit with these experimental, compact data, and we used the same Newton procedure to match this model with these data. In this case, a series of fits was done first over a range of flight conditions. These fits were then interpolated using a specific parameterization with respect to, for example, Mach number and q. The parameterization in use in this case is a quadratic of the form

$$A = A_{q2}q^2 + A_{q1}q + A_{M2}M^2 + A_{M1}M + A_{Mq}Mq + A_0. \qquad (*)$$



Experimental (curved line) and predicted (x's) flutter boundaries for BACT using approximate aeroelastic model and experimental frequency-response data. Circles in plot at right correspond to flutter points (x's) on plot at left.

Typically, the matrices in the summation have the same structure as the original aeroelastic model. However, the parameterization in Mach number and q does not necessarily reflect the one originally present in the aeroelastic model. Rather, it is to be seen as a low-order approximation of it, which is flexible enough to incorporate badly modeled parameter dependencies such as Mach number.

The plots above show the results obtained by applying this procedure to the Langley BACT data. The results are shown both on an M-q plane, and in terms of percent prediction accuracy. The latter plot requires some explanation. If the procedure is used during flight test, its accuracy becomes more important as the flight condition approaches the flutter boundary. Because the prediction is projecting to a boundary which is not as far away, the prediction also becomes more accurate as the flight condition approaches the flutter boundary. At any flight condition, as a minimum, the algorithm must say that it is safe to fly 50% of the remaining distance to the boundary. If it does not, then the envelope expansion will be halted prematurely. Overprediction of the flutter boundary may be acceptable when one is far from that boundary, but as it is approached the prediction must become more accurate.

These considerations lead to the plot at left - if the predicted flutter speed falls within the shaded region at all times, then the prediction is considered to meet the minimum requirement for usefulness during envelope expansion. For the BACT it is seen that we are very nearly successful based on only 6 data sets (this is *the minimum*

number of data sets necessary to eventually find a parameterization of the form (*)). It remains to be seen if this accuracy can be obtained for data from a real aircraft. Furthermore, since in the real case the flutter boundary will not be known, some measure of the accuracy of the prediction must be developed.

7 REMAINING RESEARCH ISSUES

Our attempts at developing the methodologies outlined above have raised the following issues:

- As compared to the BACT data used at Langley, the F-18 SRA data is more complete; however it is corrupted by higher noise levels. As a consequence, existing subspace identification procedures have difficulty providing accurate, compact representations of the data. Improving identification accuracy will require research on modifications to subspace identification methods (for instance, frequency-dependent weighting). The relationship between the efficiency of the subspace identification procedure and the excitation signal at the wing tips is also an open research question. Discussions with members of the Structures group at Dryden suggest that some alternate excitation schemes might be implementable on existing hardware (currently logarithmic and linear sine sweeps are used).

- The current nonlinear least-squares procedure to match the aeroelastic model with the 'compactified' data is computationally greedy, although specialized time-saving concepts such as Quasi-Newton have been used. While the size of the Langley system allowed the procedure to converge acceptably fast, improvements still need to be made for the same procedure to work on F-18 SRA aeroelastic models.

- The choice of an appropriate parameterization as a function of Mach number and q is still not clearly defined. The parameterization (*) is such that after optimal fitting, some entries in the matrices A_{q2}, A_{q1}, A_{M2}, A_{M1}, A_{Mq} or A_0 may not be physical (in other words, the aeroelastic model, if used alone, would predict such entries to be zero). Schemes that impose the structure of the aeroelastic are possible; however, the 'unstructured' approach allows one to correct for badly modeled parameters such as Mach number.

- In order to convert the flutter boundary problem into a robustness problem, various levels of sophistication are possible: in the most basic version, only the

Mach and q parameters appear in the perturbation block Δ. However, one can imagine that in the future, the two aforementioned steps (data compactification and least-squares model matching) provide parameterized models with error bounds; these could be easily and effectively inserted in the robustness problem as true additional uncertainties, leading to a more reliable analysis without necessarily introducing unwarranted conservatism.

- In order to make the tools usable during flight test, a significant decrease in the computation time is required. Once methods with desirable properties have been validated, optimization of the computations must be considered. Recursive implementation, for instance, should be investigated.
- An ongoing concern is proper interfacing of the proposed tool with the flight test engineer. In particular, we will need to carefully examine how data pertaining to flutter proximity should be displayed and documented in a reliable fashion. Also, any procedure able to detect malfunction of the proposed method (for example, bad fit of the experimental data) should be implemented and displayed as a warning to the operator.

8 REFERENCES

1. De Moor, B., Vandewalle, J., "A geometrical strategy for the identification of state space models of linear multivariable systems with singular value decomposition," Proc. 3rd Int. Symp. on Applications of Multivariable System Techniques, p. 59-69, Plymouth, UK, April 13-15, 1987.
2. Van Overschee, P., and De Moor, B., "A Unifying Theorem for Three Subsapce System Identification Algorithms," ESAT-SISTA Report 1993-50-J, Department of Electrical Engineering, Katholieke Universiteit Leuven, September 1994 (to appear in *Automatica*).
3. Durham, M. H., Keller, D. F., Bennett, R. M., and Wieseman, C. D., "A status report on a model for benchmark active controls testing," AIAA 91-1011, 32nd Structures, Structural Dynamics, and Materials Conference, Baltimore, MD, April 1991.
4. Gondoly, K., "Application of Advanced Robustness Analysis to Experimental Flutter," M.S. Thesis, Massachusetts Institute of Technology, June 1995.

5. Miotto, P., and Paduano, J. D., "Application of Real Structured Singular Values to Flight Control Law Validation Issues," AIAA Guidance, Navigation, and Control Conference, Baltimore, August 1995.

6. Duchesne, L., Feron, E., Paduano, J. D., and Brenner, M.,"Subspace identification with multiple data sets", 1996 AIAA Guidance, Navigation and Control Conference.

7. Bucharles, A., Cassa, H. and Roubertier, J., "Advanced parameter identification techniques for near real time flight flutter test analysis". AIAA-90-1275-CP, 1986.

8. Bryson, A. E., "Control of Spacecraft and Aircraft", Princeton University Press, 1994.