STABILITY ANALYSIS IN THE TIME-DOMAIN APPLIED TO ADAPTIVE TRANSPORT-AIRCRAFT WINGS

Wolfgang Send and Ralph Voss
Institute of Aeroelasticity, German Aerospace Center (DLR)
37073 Göttingen, Germany

Keywords: *aeroelastic stability, coupling fluid/structure, contour bump*

Abstract

A procedure for coupling fluid and structure is described, in which the structure is given by a Finite Elements model, and the fluid is computed with a flow solver for compressible flow. The tool is applied to the wing model of a modern transport aircraft, which is equipped with a small contour modification on the upper surface for drag reduction in the shock region. The aeroelastic stability is investigated in comparison to the original contour shape. Particular emphasis is laid on a proper mapping of the flow quantities into the structure and vice versa.

1 Introduction

In the run-up to even larger transport-aircraft built by the European aircraft industry, several research programs were launched in the past to develop and to assess adaptive features for transport-aircraft wings. Among others, the contour bump in the shock region and the flexible wing are in the focus of attention The bump is expected to reduce the drag in the region of shock/boundary layer interaction, the flexible wing might reduce weight of the control structures. In any of these cases, an investigation was part of the program to which extend the adaptive measures might affect the aeroelastic stability.

As an example of the theoretical approach, the paper presents the results for a contour modification (named “bump”) on the upper surface of an Airbus A340-like wing model, and describes the method and the numerical tools which have been applied. The wing model considered is related to a former program named AMP (Aeroelastic Model Program) of the French and the German aircraft industry together with the respective research establishments. The AMP wing is the wind tunnel model of a 1:25 scaled Airbus A340 wing.

In the test setup, the wing was destabilized on purpose by a torsion spring, and the flutter

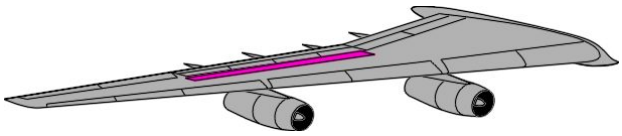


Fig. 1. Contour bump on the wing of a transport-aircraft.

characteristics were measured at fairly low Mach numbers compared to regular flight conditions.

An overview over the AMP was given by Zimmermann, Henke and Schulze [1]. A first prediction of the flutter behavior was made by the authors [2] with a bump region covering large parts of the wing span, resulting in an exaggerated sensitivity to flutter. The effect of the bump was investigated theoretically, and in a series of wind tunnel experiments, e.g. [3]. The basic benefits are the reduction of the

wave drag over a wide range of the lift and an additional effect on viscous drag for high lift. For controlling the contour of the bump several devices were developed, smart SMA structures as well as conventional tube-like circular springs. Most of the work was conducted in a joint Project ADIF (The Adaptive Wing Project) by Daimler-Chrysler Aerospace, Daimler-Chrysler Research and the DLR, the German Aerospace Center. The application of the results is not expected in the near future.

2 Fluid and Structure

The flow is computed with a 3D Full Potential code for steady and unsteady flow [4]. The code includes a 2D boundary layer correction. The mesh (Fig. 2) with 150.000 nodes is of medium size to limit the computing time. A typical result shows Fig. 3, in which a forced coupled bending and torsional motion is shown. The aerodynamic model does not include a fuselage, which was present in the wind tunnel model. The presence of the fuselage effects a slightly steeper increase of the lift coefficient c_L over the steady angle of incidence α_S, as will be seen later on.

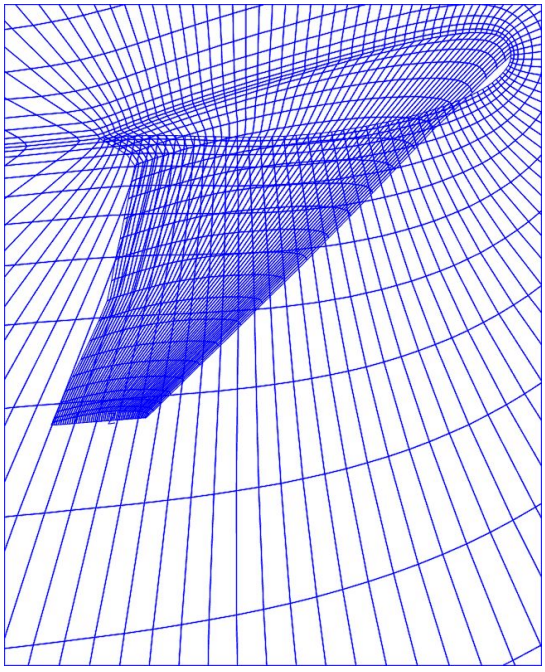


Fig. 2. The mesh applied to the AMP wing.
Mesh size: 139 x 35 x 31 nodes.

The FE model has been constructed with the

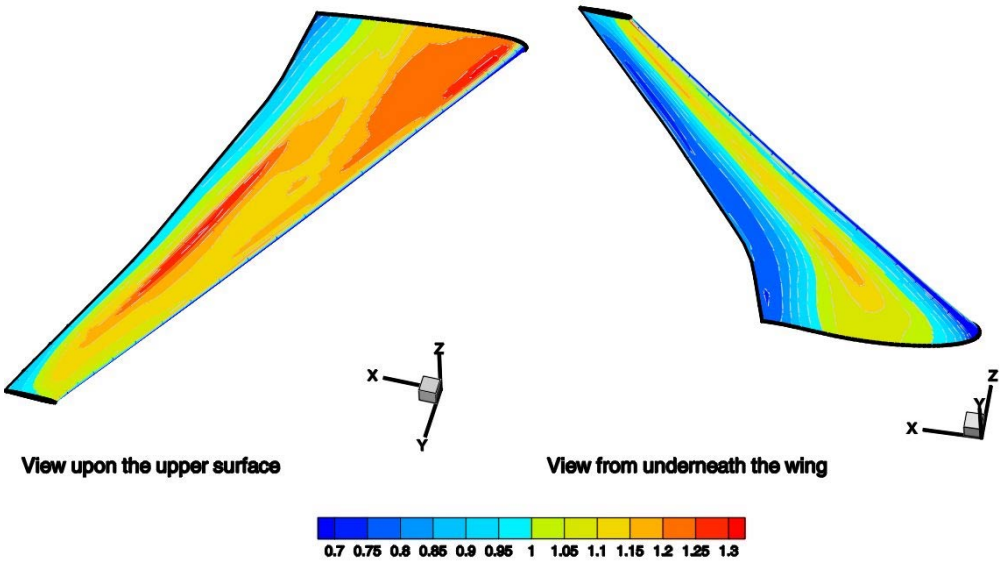


Fig. 3. AMP Wing. Local Mach numbers for a forced unsteady motion.
Two modes (bending and torsion) for Ma = 0.82, Re = 3,57 10^{6}, c_A = 0.33.

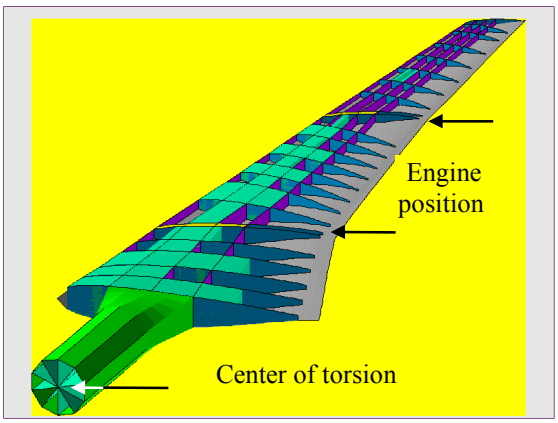


Fig. 4. Simplified FE model for the AMP wing.

ANSYS software. The surface is composed from the same surface data which generate also the aerodynamic mesh. This "generating" surface acts as an interface between the FE mesh and the aerodynamic mesh, which have different sizes and orientations of spanwise sections. Fig. 4 shows the interior of the model with the upper surface removed. The surface elements consist of a shell type, which has the capability to pick up the normal as well as the tangential forces of the fluid (Fig. 5).

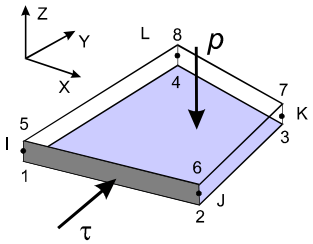


Fig. 5. Element SHELL63 with pressure and shear stress input for wing skin-paneling.

Part	Element [ANSYS]	Number	Mass [kg]
Surface	SHELL63	1200	2.689
Ribs, Spars	SHELL63	646	3.166
Interiors	SOLID73	124	6.178
Subtotal		*1970*	*12.033*
Torsion spring	SHELL63	250	1.773
Total		**2220**	**13.806**

Tab. 1. Selected properties of the FE model.

The simplified finite element model is completely constructed using aluminum as material. Tab. 1 lists some of the model's properties. The wind tunnel model was slightly heavier with a total mass of 14.48 kg, included 2.71 kg for the wing root structure with the torsion spring. The original FE model of the wind tunnel model was a beam model without any features for accessing the surface. Thus, the main purpose of designing a new model was to apply structural elements on the surface which allow to pass the fluid quantities pressure and shear stress to the structure without additional calculations on the aerodynamic side.

3 The Mapping Algorithm

The mapping algorithm provides the proper exchange of data between the physical domains, the fluid and the structure. The following part has been presented already at an earlier conference [5], however it is not yet published.

Two independent meshes are involved in the process of fluid/structure coupling, which both are derived from the wing shape: The aerodynamic mesh and the structural mesh. A mesh is generated from the node grid. The term *grid* is used in this paper to name the set of node points which put up the respective mesh.

The *wing shape* is given as a series of profiles. The J_W profile sections with I_W discrete data points $\mathbf{x}_W = (x_W, y_W, z_W)$ in each section define the wing shape W.

$$W = \left\{ \begin{array}{c} \mathbf{x}_W = [x_W(i,j), y_W(i,j,), z_W(i,j)]; \\ i = 1,..,I_W,\ j = 1,..,J_W \end{array} \right\} \quad (1)$$

Any point $\mathbf{x}_S = (x_S, y_S, z_S)$ on the wing surface S is given by an appropriate interpolation of the profile points $\mathbf{x}_W$:

$$S = \left\{ \begin{array}{c} \mathbf{x}_S = [x_S(u,v), y_S(u,v,), z_S(u,v)]; \\ u \subset [-1,+1],\ v \subset [0,1] \end{array} \right\} \quad (2)$$

u interpolates the profile from the trailing edge on the lower surface (u=-1), to the leading edge (u=0) , and back to the trailing edge on the upper surface (u=1). v ranges from the wing-root section (v=0) to the tip section (v=1).

The arrays $\mathbf{u}_W$ and $\mathbf{v}_W$

$$\begin{aligned} \mathbf{u}_W &= \{u_i;\ i = 1, \ldots I_W\} \\ \mathbf{v}_W &= \{v_j;\ j = 1, \ldots J_W\} \end{aligned} \qquad (2a)$$

are defined such that

$$W = \left\{ \begin{array}{l} \mathbf{x}_S = [x_S(u_i, v_j), y_S(u_i, v_j), z_S(u_i, v_j)]; \\ \quad i = 1, \ldots, I_W,\ j = 1, \ldots, J_W \end{array} \right. \qquad (2b)$$

Fig. 6 shows the discrete points of the wing shape with the profile sections used for the simplified AMP model. Equation (2) is used for creating both the aerodynamic grid and the structural grid. The number of sections J_W, from which the wing surface $\mathbf{x}_S$ is built up, may be much smaller.

The *aerodynamic grid* on the surface is part of the global mesh around the whole wing. J_A sections with I_A discrete data points $\mathbf{x}_A = (x_A, y_A, z_A)$ in each section define the aerodynamic grid A .

$$A = \left\{ \begin{array}{l} \mathbf{x}_A = [x_A(i,j), y_A(i,j,), z_A(i,j)]; \\ i = 1,..,I_A,\ j = 1,..,J_A \end{array} \right\} \qquad (3)$$

The set of points in Eq. (3) forms the inner

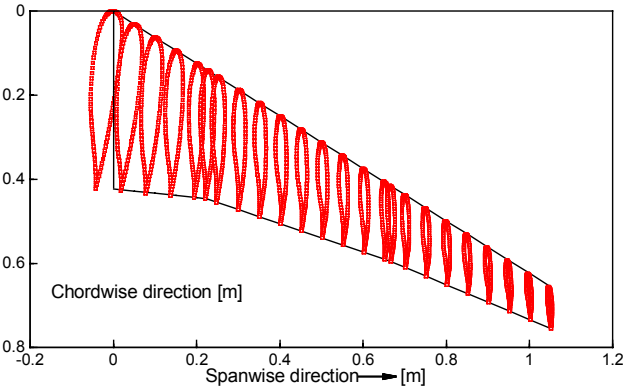


Fig. 6. Shape of the AMP wing with profile sections.

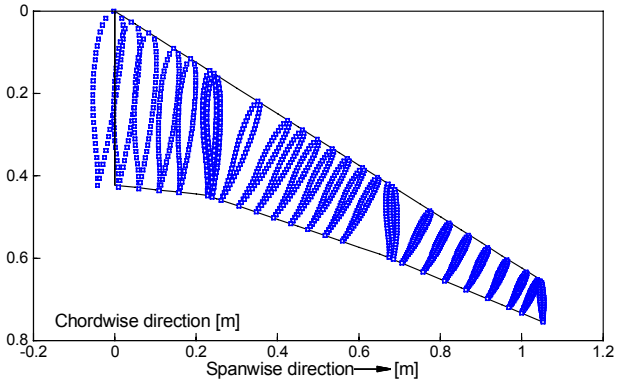


Fig.7. Nodes of structural grid for the simplified finite element model.

boundary of the CFD grid. Size and spacing have to meet the numerical requirements for the flow calculations. The grid is not depicted here. On the wing surface, it looks very similar to Fig. 6.

In general, the flow computation needs a grid with a higher node density compared to the *structural grid*, which reflects the mechanical construction of the wing. The nodes in Fig.7 are oriented to serve as key data to the finite element program. They are the corner points for the wing skin-panels, and - in the simplified model - for the ribs and spars of the wing. In both figures, Fig. 6 and Fig.7, the thickness of the sections is displayed by adding the vertical component to the spanwise coordinate. The two noticeable nearby sections in flow direction enclose the engine pylons.

The node arrangement in Fig.7 is achieved by setting the number K_N of sections, the number L_N of nodes in each section, the trailing edge position y_{NS} and the angle α_{NS} included by the respective section and the flow direction. All sections are assumed to have the same number of nodes. The geometric positions $\mathbf{x}_N$ of the nodes for the structure at rest are given by

$$\mathbf{x}_N = \begin{bmatrix} x_N(u_F(k,l), v_F(k,l)) \\ y_N(u_F(k,l), v_F(k,l)) \\ z_N(u_F(k,l), v_F(k,l)) \end{bmatrix} \quad \begin{array}{l} k = 0,..,K_N \\ l = 1,..,L_N \end{array} \qquad (4)$$

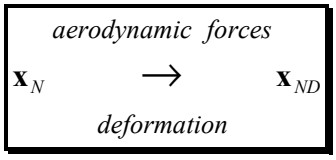
The parameter arrays $\mathbf{u}_F$ and $\mathbf{v}_F$

$$\mathbf{u}_F(k,l),\ \mathbf{v}_F(k,l) \qquad (4a)$$

are calculated using Eq. (2). The set N of node points is given in Eq. (4) and depicted in Fig.7. Notice has to be taken of the numbering: The points $k=K_N$ run along the trailing edge, the points $k=1$ are the first ones on the lower side of the wing towards the leading edge. The duplicate nodes $k=0$ are not passed to the finite element model.

4 The Posed Problem

Under the influence of the aerodynamic forces on the surface, the structure deforms (indexed by D), and the points $\mathbf{x}_N$ are moved to a different position $\mathbf{x}_{ND}$.



In which way the result of the flow calculation given for A is mapped onto N will be described later. The new positions are returned from the finite element calculation, which is an analysis for a loaded structure. The set N_D of node coordinates for the deformed structure

$$N_D = \left\{ \begin{aligned} &\mathbf{x}_{ND} = [x_{ND}(k,l), y_{ND}(k,l,), z_{ND}(k,l)]; \\ &k = 0,..,K_N,\ l = 1,..,L_N \end{aligned} \right\} \qquad (5)$$

contains the information about the position of the wing, which is the only information available. The knowledge about the position of the aerodynamic grid A refers to the position at rest, not to the deformed state. Thus, the problem to be solved is stated as:

- Where is the position of the aerodynamic grid A_D attached to the respective set N_D?

Formulated in other words, the solution of the problem is the entire reconstruction of the data for the original A and A_D from the set of nodes N and N_D, respectively. A_D denotes the set deformed points. To distinguish original and reconstructed sets, the reconstructed ones are indexed by an asterisk, i.e. A^* and A^*_D. The accuracy of the reconstructed A^* can easily be checked by comparison with the original set A, which is known, whereas an original A_D does not exist. The accuracy found for the reconstructed A^* is the measure for the accuracy to be assumed for the deformed surface A^*_D.

Several reasons count for proceeding the described way for coupling fluid and structure. As a matter of fact, flow solvers and finite element programs are separate tools, which are designed for different purposes and, in general, are developed by different people. The first ones compute normal and tangential fluid forces, pressure and shear stress, around a given surface, despite the complexity of the body considered. The latter ones determine the deformation of a structure under the influence of an external distributed load on the structure's surface.

In which way ever the tools manage to communicate properly: The fluid imposes the load, and the structure returns the deformation. Both sides are coupled to each other by the dynamic equations, which are either formulated in physical or in modal coordinates.

Besides the logical clarity of this relationship, the user might wish to apply various flow solvers or try another FE-model for the same solver. For a larger structure, which is composed of several individual substructures, the described method holds true for one surface of these substructures like fuselage, wing or engine cowling.

5 Reconstruction Procedure

The procedure uses the two-dimensional B-spline interpolation routine taken from the IMSL [6]. The well documented routine guarantees the proper set up of the algorithm:

dbsnak given m data points in the array $\mathbf{x}$ and the order of the spline k_X, the routine returns a knot

sequence **s** that is appropriate for interpolation of data on **x** by splines of order k_X.

dbs2in given two arrays **x** and **y** of dimensions m and n , two node sequences **s** and **t** of order k_X and k_Y, the routine computes a two-dimensional tensor-product B-spline interpolant for the function array **f**(**x**,**y**), and returns the tensor-product B-spline coefficients **b**.

dbs2dg for a given pair of knot sequences **s** and **t** and the B-spline coefficients **b**, the routine returns the interpolated value of the function *f(x,y)* for any x and y in the range of **x** and **y**.

Settings throughout the procedure:

$$k_X = 3,\ k_Y = 3 \tag{6}$$

The reconstruction is an iterative scheme in which the surface A^*_D is calculated from the structural grid N_D. In this paper, the scheme is derived for W^* (cf. Fig. 6) obtained from N. The procedure is the same as for getting A^* from N.

The *integer* indices k,l in Eq. (4a) identify the nodes, and the parameter arrays $\mathbf{u}_F(k,l)$, $\mathbf{v}_F(k,l)$ relate the nodes to the surface S.

The arrays are treated now as being functions of *real* arguments k and l:

$$\begin{aligned} &u_F(k,l) \ \ with \ \ k \subset [0,K_N],\ l \subset [1,L_N] \\ &v_F(k,l) \ \ with \ \ k \subset [0,K_N],\ l \subset [1,L_N] \end{aligned} \tag{7}$$

Thus, it is obvious that $W = \{\mathbf{x}_W\}$ in Eq. (1) may also be given by some values $k^*(i,j)$ and $l^*(i,j)$ such that holds true:

$$\mathbf{x}_W = \begin{bmatrix} x_W(k^*(i,j),l^*(i,j)) \\ y_W(k^*(i,j),l^*(i,j)) \\ z_W(k^*(i,j),l^*(i,j)) \end{bmatrix} \quad \begin{matrix} i = 1,..,I_W \\ j = 1,..,J_W \end{matrix} \tag{8}$$

The numbering i,j in Eq. (8) is identical with the one in Eq. (1). The two arrays

$$\begin{aligned} &\mathbf{k}_N = \{k;\ k = 0, \ldots K_N\} \ \text{and} \\ &\mathbf{l}_N = \{l;\ l = 1, \ldots L_N\} \end{aligned} \tag{9}$$

are the data arrays in the B-spline interpolation, on which the parameter arrays $\mathbf{u}_F$ and $\mathbf{v}_F$ in Eq. (4a) are interpolated. The calls

$$dbsnak(K_N+1, \mathbf{k}_N, k_X, \mathbf{s}) \tag{10a}$$

$$dbsnak(L_N, \mathbf{l}_N, k_Y, \mathbf{t}) \tag{10b}$$

$$dbs2in(K_N+1, \mathbf{k}_N, L_N, \mathbf{l}_N, \mathbf{u}_F, K_N, k_X, k_Y, \mathbf{s}, \mathbf{t}, \mathbf{b}_U) \tag{10c}$$

$$dbs2in(K_N+1, \mathbf{k}_N, L_N, \mathbf{l}_N, \mathbf{v}_F, K_N, k_X, k_Y, \mathbf{s}, \mathbf{t}, \mathbf{b}_V) \tag{10d}$$

return the spline coefficients $\mathbf{b}_U$ and $\mathbf{b}_V$ for the functions $u_F(k,l)$ and $v_F(k,l)$. This has to be done once prior to the iteration. The aim of the iteration is finding $k^*(i,j)$ and $l^*(i,j)$ such that

$$\begin{aligned} &\left|u(k^*(i,j),l^*(i,j)) - u_i\right| < \varepsilon \ \text{and} \\ &\left|v(k^*(i,j),l^*(i,j)) - v_j\right| < \varepsilon \ \text{for all } i,j \end{aligned} \tag{11}$$

with ε being a small number.

6 Global Iteration

The global iteration repeats the following local iterations until the differences in Eq. (11) fall short of the limit set, or the maximum number of global iterations is exceeded. At present, the author has not yet gained much experience with different configurations. Those which have been investigated converge very fast within a few steps. The result for the AMP wing is given at the end of this paragraph.

7 Local Iteration

The local iteration is done in two stages, in which both times the procedure steps through all points of the wing arrays $\mathbf{u}_W$ and $\mathbf{v}_W$. In the first stage the spanwise positions are estimated

by applying a local Newton iteration, in the second stage the circumferential positions are treated the same way using the previously calculated spanwise position. The result at the end of the second stage is subject to the poll in Eq. (11).

8 Local Iteration: First Stage

The outer loop steps through the profile sections from the root to the tip, i.e. j=1, ...,J_W. A first guess for l^* is

$$l^*_{ij} = 1 + v_j \cdot (L_N - 1). \tag{12}$$

The inner loop runs in circumferential direction i=1, ...,I_W. The initial guess for k^* is

$$k^*_{ij} = 1 + K_N / 2 \cdot (u_i + 1). \tag{13}$$

in the first global iteration. In the following global iterations the value k^* is taken from the previous second stage. For each k^* in the inner loop the l^* is iterated to

$$|g_j| < \varepsilon \text{ with} \tag{14}$$
$$g_j = v(k^*(i,j), l^*) - v_j$$

with a Newton iteration for

$$l^*_{n+1} = l^*_n - g_j(k^*, l^*_n) / (\partial g_j(k^*, l^*_n) / \partial l^*_n). \tag{15}$$

The iteration ends in the way as the global does. The routine *dbs2dg* provides the interpolated function and its partial derivatives. The call for v reads (without the option for multiple call):

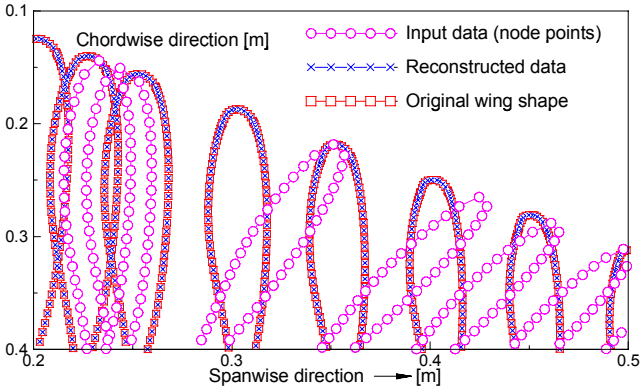


Fig. 8. Reconstruction of W^* from the set N of node points; accuracy is demonstrated by comparison to the original wing shape W.

Global it.	Max. number local it. for l^*	Max. number local it. for k^*	Global error
1	8	3	0.113
2	8	2	1.22d-03
3	8	2	2.65d-05
4	8	1	6.85d-07
5	8	1	1.83d-08
6	8	1	4.92d-10
7	8	1	9.98d-11

Tab. 2. Figures of iteration procedure for ε = 1d-10.

$$dbs2dg(p,q,k^*,l^*,K_N+1,L_N, \tag{16}$$
$$k_X, k_Y, \mathbf{s}, \mathbf{t}, \mathbf{b}_V, v_F(p,q))$$

p and q denote the order of the partial derivatives with respect to k^* and l^*. The returned function depends on the choice of p,q. At the end of the first stage the arrays $\mathbf{k}^*_N$ and $\mathbf{l}^*_N$ are passed to the next stage as inital values.

9 Local Iteration: Second Stage

Here, the outer loop steps in circumferential direction i=1, ...,I_W, and the inner loop runs through the profile sections from the root to the tip, i.e. j=1, ...,J_W. k^* is iterated to

$$|h_i| < \varepsilon \text{ with} \tag{17}$$
$$h_i = u(k^*, l^*(i,j)) - u_i$$

with a Newton iteration for

$$k^*_{n+1} = k^*_n - h_i(k^*_n, l^*) / (\partial h_{ji}(k^{**}_n, l^*) / \partial k^*_n). \tag{18}$$

The call for u reads:

$$dbs2dg(p,q,k^*,l^*,K_N+1,L_N, \tag{19}$$
$$k_X, k_Y, \mathbf{s}, \mathbf{t}, \mathbf{b}_U, u_F(p,q))$$

Since in each stage only one variable is used for the fit, the global result for both variables converges slower than the individual local iterations do.

10 Result for the Reconstruction

Tab. 2 shows the figures during the reconstruction W^* of the wing shape W in Fig. 6 from the node set N in Fig.7. The maximum

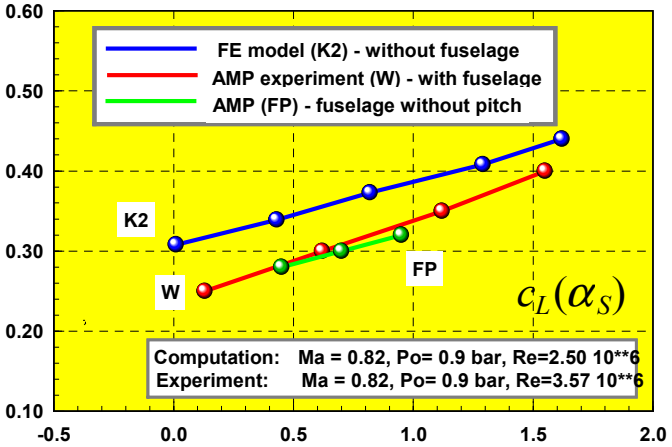


Fig. 9. Results for static coupling in comparison with wind tunnel experiments. Lift coefficient versus angle of incidence.

number of local iterations for eqs. (15) and (18) are given in the two columns in the middle. The global error refers to Eq. (11). The final result

$$\mathbf{k}_N^* = \left\{k_{ij}^*; i = 1, \ldots I_W, j = 1, \ldots J_W\right\} \text{ and}$$
$$\mathbf{l}_N^* = \left\{l_{ij}^*; i = 1, \ldots I_W, j = 1, \ldots J_W\right\} \qquad (20)$$

is used to demonstrate the accuracy of the reconstruction. Fig. 8 shows the comparison with the orginal wing shape for one arbitrary section of Fig. 6, which is clipped to blow up the details.

- The result of the reconstruction is a precise knowledge of the wing shape from the positions of the node points.
- The position of the deformed aero-dynamic mesh is assumed to be known with the same accuracy.

11 Static Coupling

The results for the static coupling are compared to wind tunnel experiments. Unfortunately, the global data are taken from reports which are non public available. The data lift versus angle of incidence in Fig. 9 refer to an experimental model with fuselage, in which the fuselage could be kept at rest separately from the wing. The few green data (FP) are the wing turned without

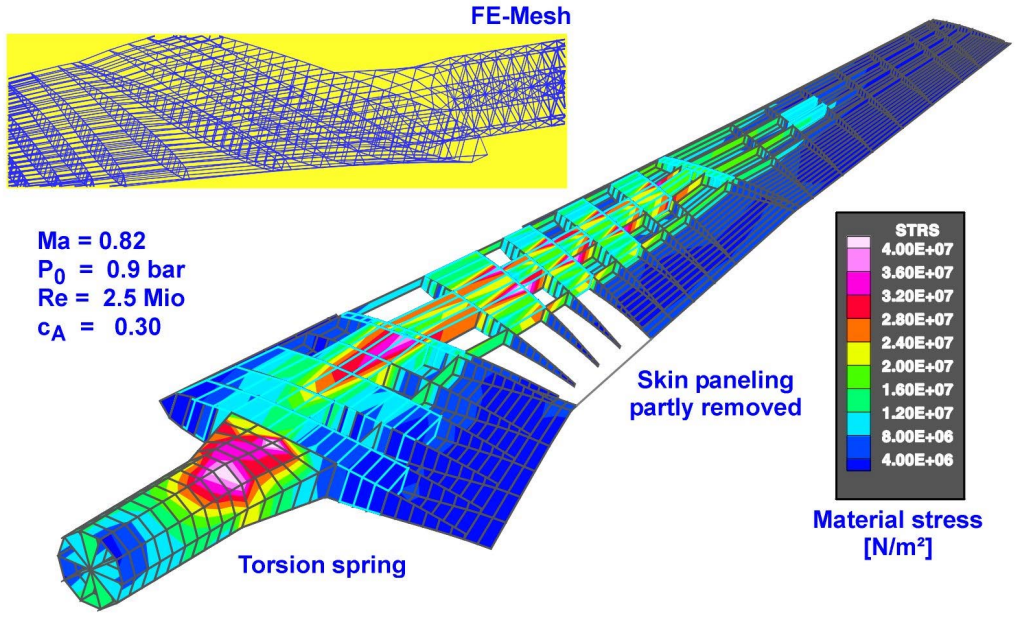


Fig. 10. Static coupling of fluid and structure. Stress distribution in the simplified AMP FE model.

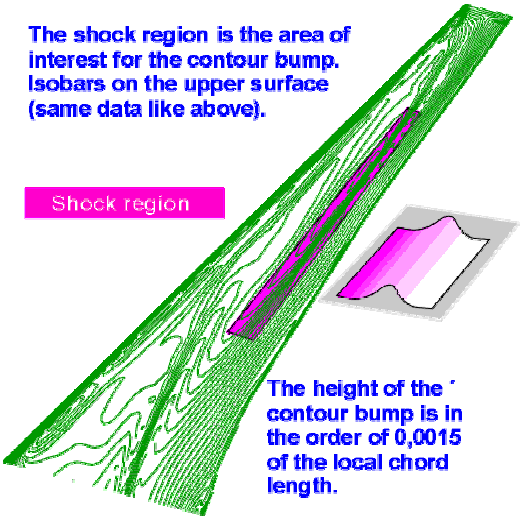


Fig. 11. The contour modification by a „bump“ on the upper side of the wing.

the fuselage. They show the same slope as the theoretical data. The red dots (W) are the whole model inclined. The slope is slightly higher. In general differ the theoretical data (computed for free stream conditions) from the wind tunnel data by about 0.7 degree. To each converged solution belongs the full information about the structural data. In Fig. 10 the stress distribution is shown for $c_L = 0.3$ and $Ma = 0.82$.

12 Coupling for Unsteady Flow

The new FE model was needed for the new strategy of coupling fluid and structure. Delicate structures like the contour bump (Fig. 11) require full control of the coupling process between the aerodynamic mesh and the structural mesh.

The stability analysis can be carried out with various sets of natural modes, which is time consuming even on the newest powerful computers. The modes are computed in the FE model. The mode shapes are mapped into the surface of the CFD code. The minimum set is a model with N=2 degrees of freedom. The reduction leads to a mass matrix **M** and a stiffness matrix **K** of the desired order N. The first two modes are the ones which couple in the flutter experiment. The selected modes are expanded again (depending on how the model has been reduced), and the respective mode shapes provide the information needed for the CFD analysis. The CFD code delivers the fluid force **f** at any time step.

The system of N differential equations

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{K}\mathbf{q} = -\mathbf{f}(\dot{\mathbf{q}}, \mathbf{q}, t) \qquad (21)$$

is reduced to a first order system of $2N$ equations. The **q** are the generalized coordinates for the individual mode shapes, one function $q(t)$ for each mode. The system is solved by the documented IMSL routine *divpag*. The routine offers two classes of

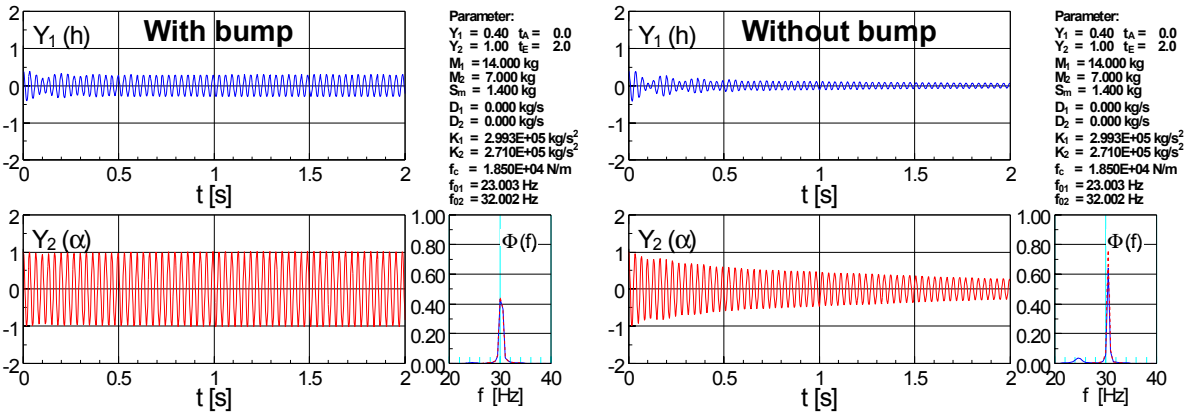


Fig. 12. Integration in the time domain with the first two modes „First Bending“ and „Rigid Torsion“.

implicit linear multistep methods: The Adams-Moulton method or Gear's backward differentiation formulas. The process of stepping back in time is carefully watched, and the flow history is monitored. However to the authors' experience, the proper integration of the structural part, regarding reliable phase relations among the DOFs, requires the small time steps needed in the integration process, rather than the physics of the flow field.

Fig. 12 shows a typical result. Compared to the configuration without bump, the bump causes a slight destabilization, which does not affect seriously the aeroelastic stability.

In an earlier paper, the authors had published results with more dramatic changes in the stability limits. The results were based on a similar 2-DOFs model like the one shown here. However, at that time neither the converged solution of the distorted wing had been included, nor the modes of a real structure were used. Instead, the two modes had been rigid modes for the plunge as well as for pitch.

The findings are confirmed by experimental investigations with 2D profiles, where variations of the flutter boundary are within the scattering of the results by other flow phenomena like forced and free transition.

Acknowledgments

The authors appreciate the financial support which was organized by the ADIF project management.

References

[1] Zimmermann, H, Henke, H and Schulze, B. *Computation of Flutter Boundaries in the Time and Frequency Domain*, AGARD Structures and Materials Panel, Specialists' Meeting on Transonic Unsteady Aerodynamics and Aeroelasticity, San Diego, CA, 0ctober 9-11, 1991.

[2] Send, W and Voß, R. *Effects of a Contour Bump on the Transonic Aerodynamic Forces of the AMP Wing.* Notes on Numerical Fluid Mechanics, Vol. 72: New Results in Numerical and Experimental Fluid Mechanics II. Vieweg, Braunschweig 1998, ISBN 3-528-03122-0.

[3] Birkemeyer, J. *Widerstandsverminderung für den transsonischen Flügel durch Ventilation und adaptive Konturbeule.* DLR FB 1999-28. Köln 1999. ISSN 1434-8454.

[4] Lu, Z and Voß, R. *DLR FP+VII Code Improvements and Computation of Unsteady Transonic Flow*. DLR IB 232 - 96 J03, Göttingen 1996.

[5] Send, W. *Coupling of Fluid and Structure for Transport Aircraft Wings*. Int. Forum on Aeroelasticity and Structural Dynamics, Williamsburg, VA, June 22-25, 1999.

[6] IMSL Math/Library, Visual Numerics Inc., 6230 Lookout Road, Boulder, Colorado 80301, USA