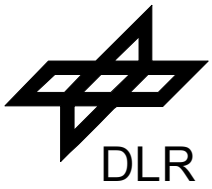
German Aerospace Center



Coupling of Fluid and Structure for Transport Aircraft Wings

Wolfgang Send
DLR Institute of Aeroelasticity, Göttingen, Germany

International Forum on Aeroelasticity
and Structural Dynamics
CEAS / AIAA / ICASE / NASA Langley
Williamsburg, Virginia
June 22-25, 1999

Deutsches Zentrum für Luft- und Raumfahrt e.V.

DLR-Institut für Aeroelastik
D-37073 Göttingen, Germany

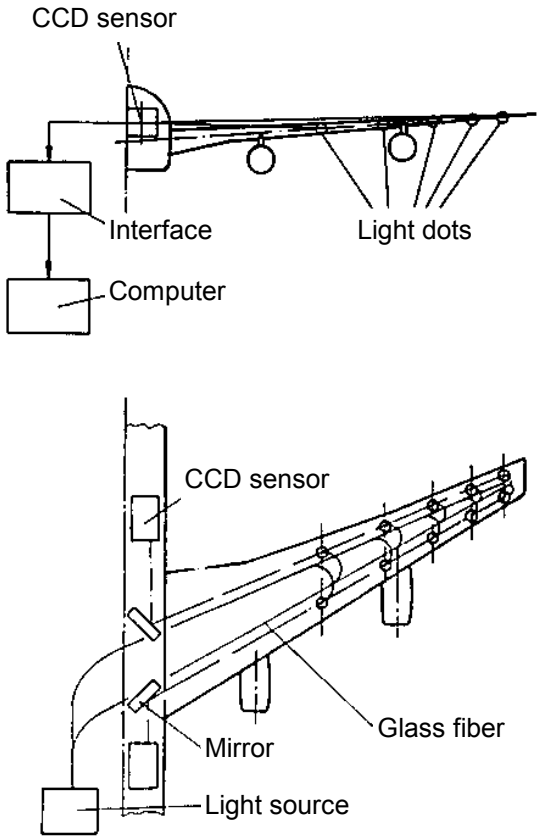
June 1999

Abstract

A simplified finite element model has been constructed to study the techniques for coupling transonic flow around transport aircraft wings and their structural response. The investigation is stimulated by the current considerations to introduce various adaptive structures into the wing design of large transport aircraft. The procedure including a two-dimensional B-spline interpolation is described by which the deflected surface of the finite element model is recalculated to the size of the aerodynamic CFD grid with the same deformation. The flow is computed using a full potential code with boundary layer coupling.

Introduction

Within the scope of the concept „Adaptiver Flügel“ (Adaptive Wing) by Daimler-Chrysler Aerospace, Daimler-Chrysler Forschung and DLR, the current research considers to introduce various adaptive structures into the wing design of large transport aircraft. One of the features, the *contour bump*, pursues the aim of reducing drag by adapting a small region of the upper wing contour to transonic flow conditions. Research also comprises studies on side effects like altered aeroelastic stability as a consequence of adaptive measures. To study the effects on 3D surfaces, the AMP wing (Aeroelastic Model Program) has been chosen as an appropriate model for modern transport aircraft wings. Among others, results from the AMP experiments in 1987-1991 were presented by *Deutsche Airbus* [1]. The AMP wing is the windtunnel model of a 1:25 scaled Airbus A340 wing.



The AMP windtunnel model

A simplified finite element model of the AMP wing has been constructed to study the techniques of coupling transonic flow around aircraft wings and their structural response on the basis of a certified finite element tool. First results from a coupling computation in steady flow are shown. The final goal is the development of the *numerical flight test* for a wing (and in the future for the complete aircraft).

Wing Geometry

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.

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

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 $\mathbf{x}_W$:

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

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

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

Figure 1 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 .

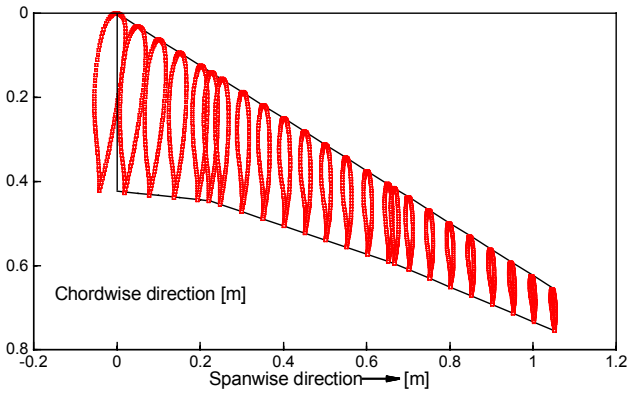


Fig. 1: Shape of the AMP wing with profile sections.

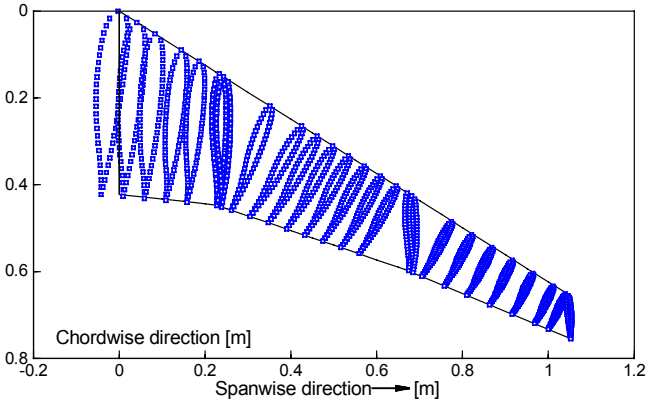


Fig. 2: Nodes of structural grid for the simplified finite element model.

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

The set of points in eq. (3) forms the inner 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. 1.

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. 2 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 1 and 2, 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. 2 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 β_{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

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

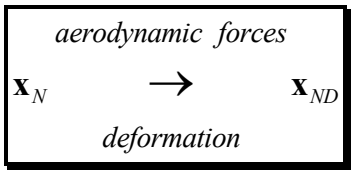
The parameter arrays $\mathbf{u}_F$ and $\mathbf{v}_F$

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

are calculated using eq. (2). The set N of node points is given in eq. (4) and depicted in fig. 2. 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.

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

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

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 „point of sale" is the surface, no other place. 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.

Reconstruction Procedure

The procedure uses the two-dimensional B-spline interpolation routine taken from the IMSL [2]. The well documented routine guarantees the proper set up of the algorithm:

DBSNAK	given m data points in the array **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. 1) 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:

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

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

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

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

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

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

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

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

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.

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.

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).

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

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

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

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

with a Newton iteration for

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

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):

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

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.

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

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

with a Newton iteration for

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

The call for u reads:

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

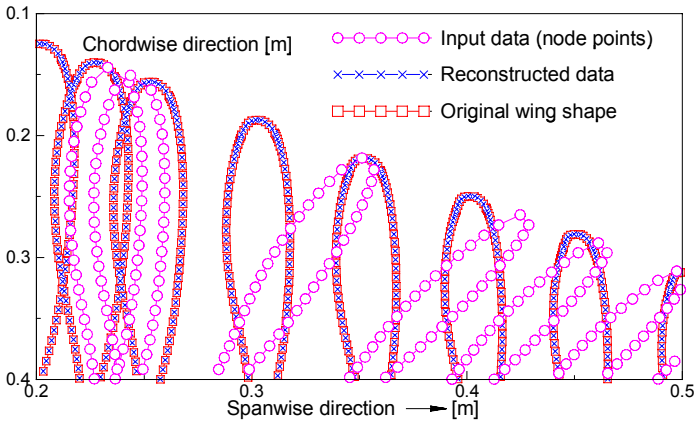


Fig. 3: Reconstruction of W^* from the set N of node points; accuracy is demonstrated by comparison to the original wing shape W.

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

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.

Result

Table 1 shows the figures during the reconstruction W^* of the wing shape W in fig. 1 from the node set N in fig. 2. The maximum 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

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

is used to demonstrate the accuracy of the reconstruction. Fig. 3 shows the comparison with the orginal wing shape for one arbitrary section of fig. 1, 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 aerodynamic mesh is assumed to be known with the same accuracy.

Finite Element Model

The simplified finite element model is completely constructed using aluminum as material. Table 2 lists some of the model's properties.

Part	Element [ANSYS]	Num-ber	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**

Table 2: Selected properties of the FE model.

The windtunnel 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. For the design of the model the FE tool ANSYS [3] is used. The original FE model of the windtunnel model is 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 subcalculations on the aerodynamics side. The element SHELL63 used for the wing skin-paneling is shown in fig. 4.

Bending stiffness and torsional stiffness of the FE model are compared with the windtunnel model by loading 100 N on the wing tip and applying a torque of 1Nm. Figure 5 shows the comparison of the two tests. The bending line of the FE model is computed from

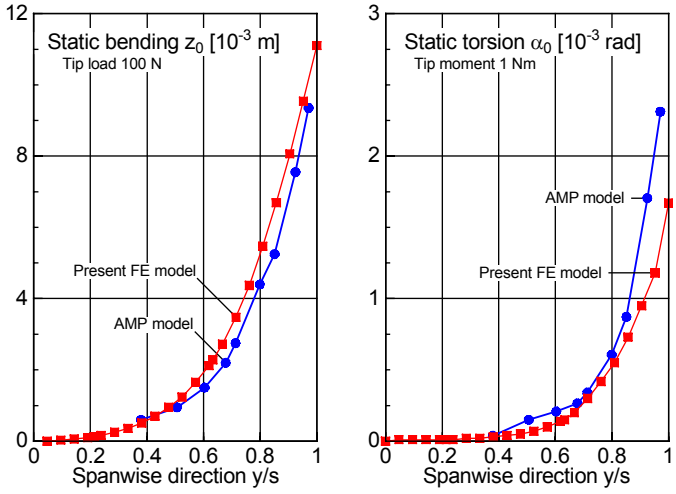


Fig. 5: Static deflections of the AMP windtunnel model and of the present FE model.

$$z_0(y) = (z_{LE}(y) + z_{TE}(y))/2\,. \qquad (21a)$$

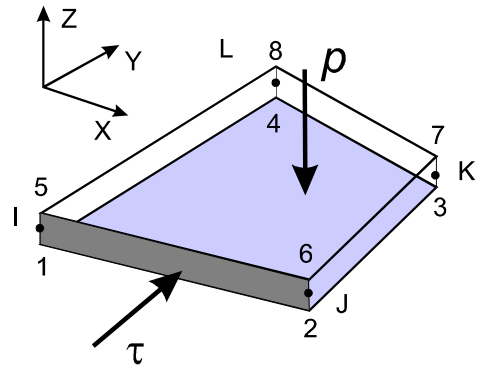


Fig. 4: Element SHELL63 with pressure and shear stress input for wing skin-paneling.

The torsion is defined by

$$\alpha_0(y) = (z_{LE}(y) - z_{TE}(y))/\ell(y)\,. \qquad (21b)$$

The indices *LE* and *TE* refer to the leading and the trailing edge, respectively. $\ell(y)$ is the local chord length. While the bending shows satisfactory

No.	Form	Exp. [Hz]	Model [Hz]
1	First bending	23.4	23.2
2	First torsion	31.8	32.3
3	1. In-plane bending	49.0	56.1
4	Second bending	55.0	69.7

Table 3: The first dynamic modes of the AMP wing

Table 3 presents the first eigenfrequencies from a modal analysis in comparison with the experimental data. The first two modes, which are used for the flutter analysis, are well met. The higher the modes are, the larger are the discrepancies. This is mainly a consequence from the simplified modelling. Particularly, the flutter brake with a spring-driven mass mounted on small boom in the experimental model, is omitted in the present FE model.

The interior of the FE model is shown in fig. 6. Due to the torsion spring, the first torsional mode with an eigenfrequency tuned to a value slightly above the first bending, is almost a rigid body mode. The small distance between these two frequencies is the reason, why the model flutters at all. The study of flutter behavior was one major research topics in the program.

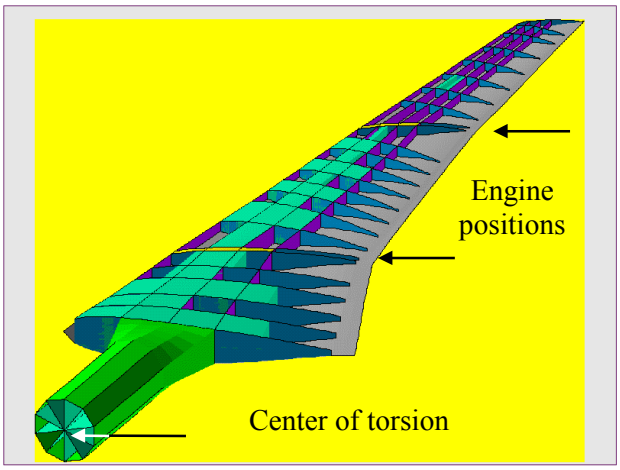


Fig. 6: Interior of the simplified FE model for the AMP wing suspended from a torsion spring.

Aerodynamics

As stated already, the coupling procedure is compatible with any aerodynamics code which provides pressure (occasionally shear stress) on a surface grid similar to the one discussed here. For the coupling procedure described in the next paragraph, a full potential code with boundary layer correction has been applied. The code may be applied for steady and unsteady computations. The code developed by Voß [4] was subject to further improvements by Lu and Voß [5].

Mach number	0.82	[-]
Reynolds number	2.5 10^6	[-]
Stagnation pressure	0.9	[bar]
Lift coefficient	0.308	[-]

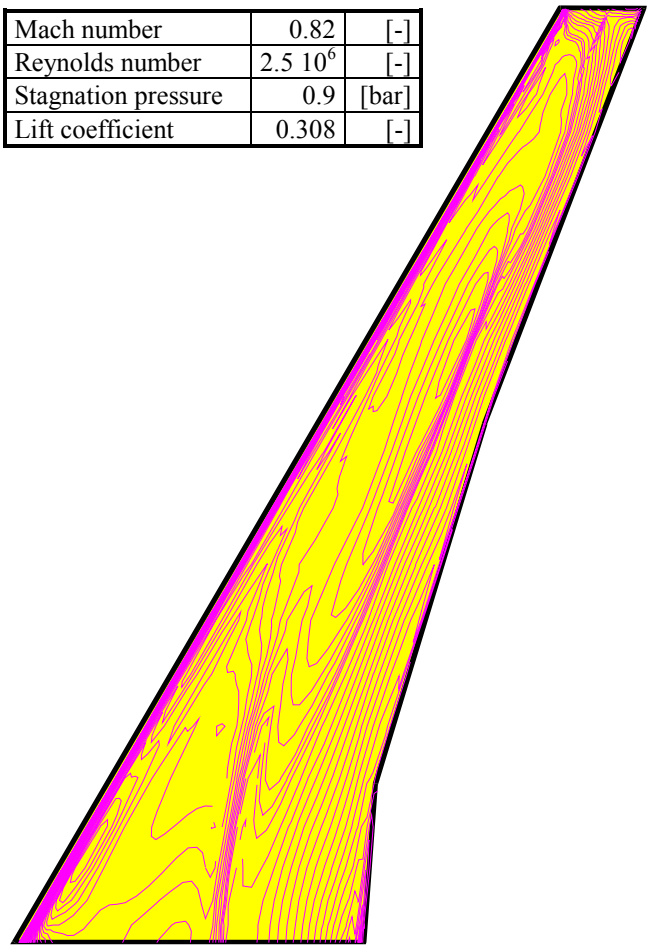


Fig. 8: Contour plot for isobars (upper side); result from fluid/structure coupling.

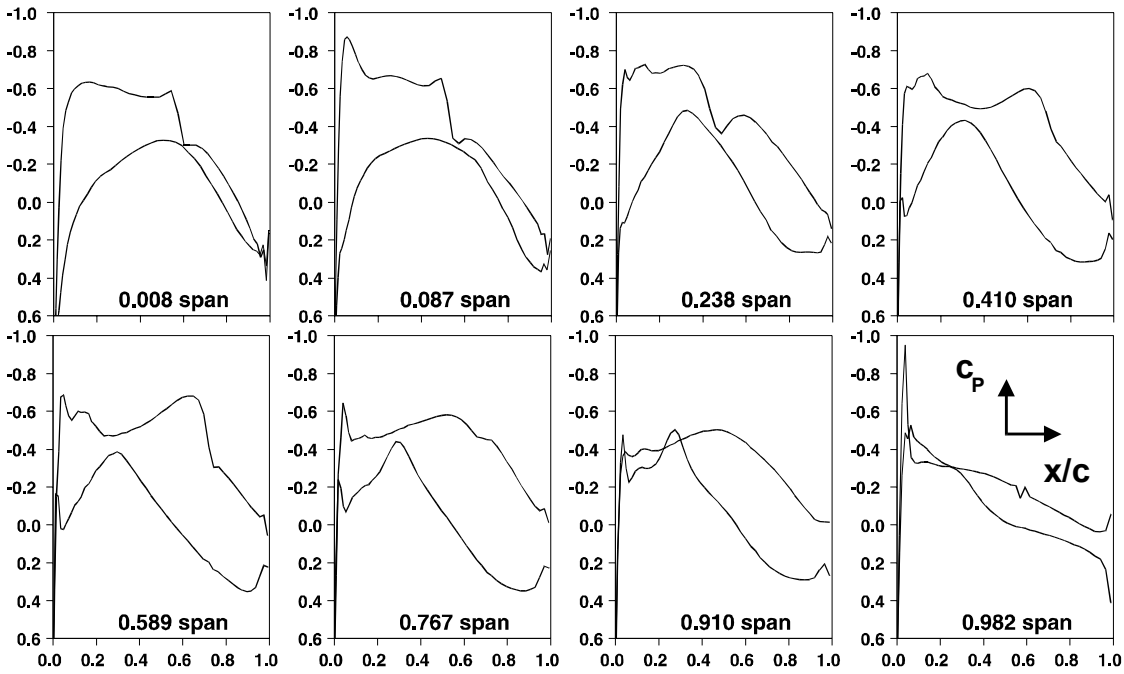


Fig. 7: Pressure distributions for various spanwise locations; flow parameters cf. fig. 8.

Coupling of Fluid and Structure

The procedure of coupling fluid and structure is illustrated in fig. 9. The first steps beginning with the generation of the aerodynamic and the structural meshes have already been discussed in the previous paragraphs.

The pressure is also interpolated with a B-spline interpolation, which guarantees the best results. Currently, no care is taken to align the aerodynamic *lift* with the structural *reaction force*. Both are supposed to be equal theoretically. The appearing differences between the two forces are in the order of a few percent or less, depending on the lift coefficient. A thorough search for the origin of the difference needs still to be done. One reason is probably the loss of information during the download of the pressure from the aerodynamic mesh to the structural mesh.

The pressure imposed on the structure is subject to a static analysis, which returns the deformation

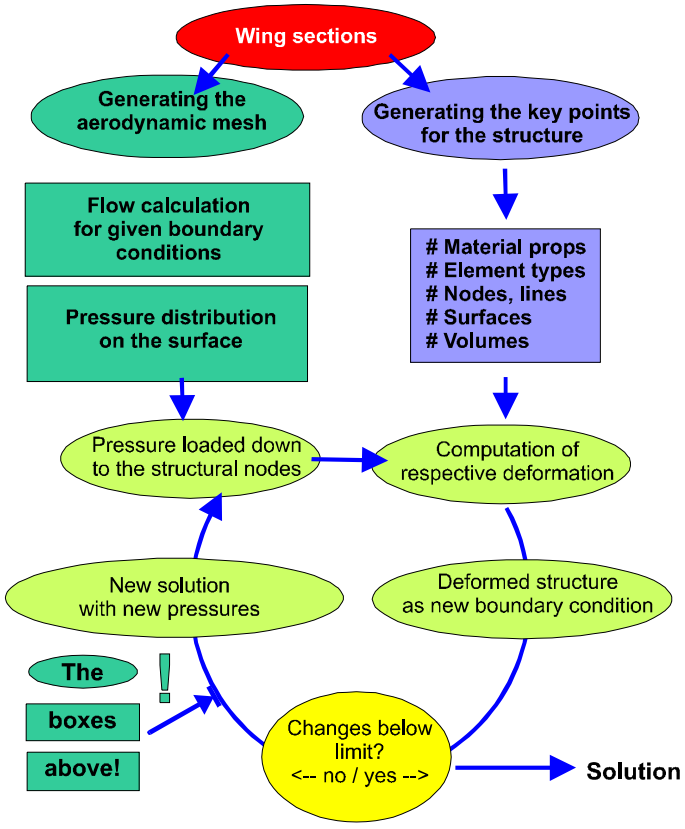
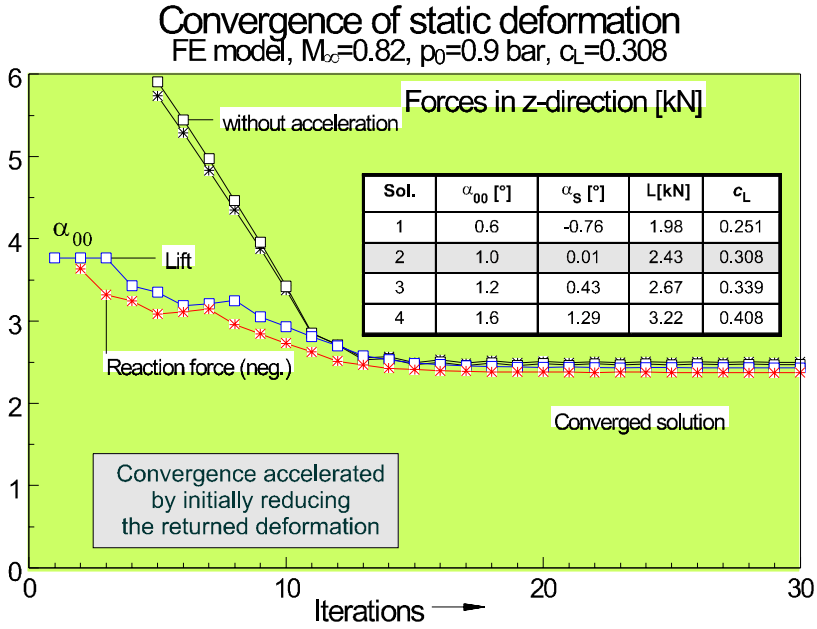


Fig. 9: Scheme for coupling of fluid and structure



Sol.	α_{00} [°]	α_s [°]	L[kN]	c_L
1	0.6	-0.76	1.98	0.251
2	1.0	0.01	2.43	0.308
3	1.2	0.43	2.67	0.339
4	1.6	1.29	3.22	0.408

Fig. 10: Coupling of fluid and structure; convergence of the procedure for steady flow and static deformation.

of the FE model. The *reconstruction procedure* finally leads to a new boundary condition for the next flow solution, which alters the pressure on the FE model. The iteration cycles through these steps until the solution is converged.

A typical record of an iteration is depicted in fig. 10. The corresponding aerodynamic solution for c_L=0.3 is shown in figs. 7 and 8, in which the isobars on the upper surface and the pressure distributions for selected sections are given.

The torsion spring experiences a sudden negative pitch at the very beginning of an iteration, which is caused by the unbalanced moment of the pressure distribution for low lift coefficients. The larger the lift is, the smaller is the pitching moment. This can be seen from table in fig. 10. To achieve an angle of incidence α_0=0 deg, the initial angle has to be set to α_{00}=1 deg for c_L=0.3. That difference of 1 deg reduces to 0.3 deg for c_L=0.4. The undesired effects on the flow calculation can be alleviated by reducing the returned deformation within the first steps of the iteration. The result of this measure is shown by the lift and reaction force curves, which are computed without 'acceleration' of the convergence.

The coupling leads to a deformation of the model, which - among others - may be decomposed into bending and torsion of the wing using eq. (21).

Figure 11 shows the variation over span for the four lift coefficients in fig. 10. The range for both bending and torsion is almost the same as for the windtunnel experiment. Unfortunately, these data are still classified and cannot be displayed here. The plot reveals a major discrepancy for the outer parts of the torsion. Whereas the windtunnel model shows the same parabolic behavior for the torsion like for the bending, the FE model differs in the torsion. The reason for the difference is probably to be found in the different torsional stiffness, which is obvious from fig. 5.

The result for the lift coefficient in fig. 12 gives an impression of the overall behavior of the coupling procedure. The smaller $d\,c_L/d\,\alpha$ for the FE model results from the missing fuselage, which contributes considerably to the lift with increasing angle of incidence. The proof is found in the experimental data, which show a similarly smaller $d\,c_L/d\,\alpha$ when the fuselage is kept at rest with increasing α . A satisfactory explanation for the difference of up to 0.5 deg in α for the same c_L is not yet found.

Concluding Remarks

The paper proves the ability of the proposed procedure for the coupling of fluid and structure in steady flow. Currently, the procedure is extended to the unsteady case. The approach to the unsteady case solves the problem in advance to relate the the aerodynamic forces to a different structural mesh and vice versa.

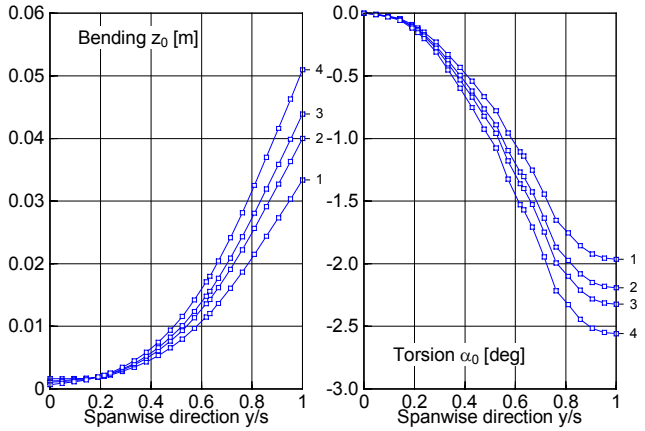


Fig. 11: Elastic deformation of the FE model for four different lift coefficients (solutions 1-4 in fig. 10).

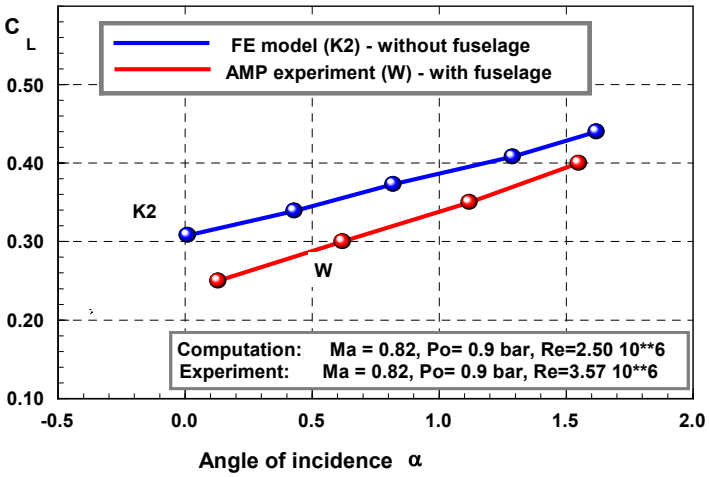


Fig. 12: Result for the lift coefficient in comparison with the windtunnel experiments

Acknowledgements

The author gratefully acknowledges the kind assistance of his colleague Dr. R. Voß during the phase of implementing the aerodynamic codes into the coupling procedure.

References

[1] Zimmermann, H., Vogel, S., Henke, H., 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] IMSL Math/Library, Visual Numerics Inc., 6230 Lookout Road, Boulder, Colorado 80301, USA

[3] ANSYS Inc., 201 Johnson Road, Houston, Pennsylvania 15342-1300, USA

[4] R. Voß, W. Wegner, D. Neisius, Computation of 2D and 3D Unsteady Transonic Flows by Inviscid and Viscous-Inviscid Interaction Methods, in: ECARP - European Computational Aerodynamics Project: Validation of CFD Codes and Assessment of Turbulence Models, Friedr. Vieweg & Sohn, Braunschweig 1997, ISBN 3-528-06958-9

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