AIAA Journal
Vol. 48, No. 11, November 2010

Harmonic-Balance-Based Code-Coupling Algorithm for Aeroelastic Systems Subjected to Forced Excitation

F. Blanc*
Airbus S.A.S., 31060 Toulouse, France
F.-X. Roux†
ONERA, 92322 Chatillon, France
and
J.-C. Jouhaud‡
Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique, 31057 Toulouse, France

DOI: 10.2514/1.45444

Aeroelastic systems may exhibit vibrations that are induced by a forced periodic motion of their elements. For example, the forced periodic motion of an aileron on a deformable wing produces vibrations of this wing. Accurate numerical simulation of such problems is usually based on a time-marching scheme and a code-coupling strategy. Such computations are costly, since before the periodic-vibration cycles, a long transient has to be computed. A technique called time-harmonic balance has been developed in the field of computational fluid dynamics to accelerate the computation of unsteady periodic flows. This technique allows the periodic state of these flows to be computed directly without computing their transients. In this paper the time-harmonic-balance strategy is applied to an aeroelastic solver consisting of two coupled codes. The time-harmonic-balance technique allows the resulting aeroelastic solver to compute the periodic-vibration cycles caused by a periodic forced motion directly, without computing the long transients. The time-harmonic-balance-based aeroelastic solver is analyzed and validated on a two-dimensional aeroelastic pitch-plunge airfoil undergoing a forced pitching. Furthermore, application to an industrial case is considered, where the flexibility of an aircraft wing is taken into account in the computation of unsteady aerodynamic forces caused by an oscillating aileron.

Nomenclature

C	=	structure damping
c_i	=	generalized damping of eigenmode i
D	=	time-harmonic-balance derivation matrix
D_k	=	time-harmonic-balance derivation operator of instant k
$F(W)$	=	vector of forces applied by the fluid on the structure
f_i	=	generalized force corresponding to eigenmode i
g_T	=	time T periodic function
I	=	identity matrix
j	=	$\sqrt{-1}$
K	=	structure stiffness
k_i	=	generalized stiffness of eigenmode i
M	=	structure inertia
m_i	=	generalized mass of eigenmode i
N_h	=	number of computed harmonics
N_s	=	number of structural eigenmodes
N_t	=	number of time steps per period
R	=	residual vector of Navier–Stokes equations
W	=	vector of fluid conservative variables
w_n	=	vector of conservative variable instant n
x^m	=	coordinates of the mesh used for spatial discretization of Navier–Stokes equations
x^s	=	coordinates of the points of the structure
α_r	=	static coupling relaxation parameter
ρ	=	fluid density
$\hat{w}_n$	=	nth complex coefficient of discrete Fourier transform of W

Subscripts

i	=	identification of structural mode
k	=	identification of time steps

I. Introduction

Wing flutter is a well-known aeroelastic phenomenon that can cause serious damage or even destroy an aircraft when a certain limit speed is exceeded [1,2]. However, in an industrial context there are many cases in which the vibrations of an aeroelastic system are not self-induced, but caused by an external periodic excitation. Some examples of such phenomena are vibration of turbomachine blades induced by periodic crossing of other blade wakes, oscillatory motion of an aircraft wing caused by the periodic motion of a control surface, or periodic deformation of a helicopter blade triggered by cyclic controls.

The accurate numerical simulation of these phenomena is of great industrial interest and it is usually based on a code-coupling strategy. Computation of the periodic-vibration cycle is done by running the computation and waiting for the end of all transients to obtain the periodic state of the system. Transients can be very long and such strategies may lead to very high computation times, making such simulations difficult to use in an industrial context.

This long transient issue is also encountered in nonaeroelastic pure fluid dynamics problems: the computation of time-periodic flows with a usual time-domain-written computational fluid dynamics (CFD) solver requires processing a transient flow as a first and time-consuming step. To solve this, a method called time-harmonic balance (THB) has been developed in the field of CFD for

Received 14 May 2009; revision received 2 March 2010; accepted for publication 16 July 2010.

*Ph.D. Student, Aerodynamics Department; Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique, 316 route de Bayonne; florian.blanc@airbus.com.

†Senior Researcher, Modeling and Information Processing, High Performance Computing Unit, B.P. 72, 29 Avenue de la Division Leclerc.

‡Senior Researcher, Advanced Aerodynamics and Multiphysics Team, 42 Avenue Gaspard Coriolis.

accelerating the computation of periodic flows. Using this technique it is possible to directly compute the periodic state of a flow without computing its transient, leading to a significant reduction of computation times. The principle of THB is to transform a costly unsteady fluid dynamics computation into several steady computations running in parallel and coupled by a source term.

In this paper, an extension of THB of pure CFD to a code-coupling based aeroelastic solver is proposed. With this extension, it is possible to directly compute the periodic-vibration cycles caused by the periodic forced motion of one part of an aeroelastic system. As with pure CFD problems, THB transforms the unsteady aeroelastic computation into several static and coupled aeroelastic simulations.

In the first part of this paper, a generic aeroelastic problem is presented and the usual methods for solving it are briefly described. In the second part, the THB technique of CFD and its extension to an aeroelastic solver is proposed. In the third part, the method is validated on a simple case by comparing the results with the THB-based code-coupling strategy to those obtained with a state-of-the-art code-coupling algorithm. Finally, the method is applied to compute the vibration cycle caused by the periodic motion of an aileron on an aircraft wing.

II. Existing Methods for Computing Aeroelastic System Response to Forced Excitation

A. Generic Equations of an Aeroelastic Problem

The generic equations of an aeroelastic problem with a forced motion of a part of the system can be written in the following form:

$$\frac{\partial W}{\partial t} = R\left(W, \frac{\mathrm{d}x^m}{\mathrm{d}t}, x^m\right) \tag{1}$$

$$\mathrm{Struct}\left(\frac{\mathrm{d}^2 x^s}{\mathrm{d}t^2}, \frac{\mathrm{d}x^s}{\mathrm{d}t}, x^s, F(W)\right) = 0 \tag{2}$$

$$x^m = \mathrm{MeshDef}(x^s) \tag{3}$$

$$\exists i \text{ such that: } x^s[i] = g_T(t) \tag{4}$$

1) Eq. (1) is the conservative form of the Navier–Stokes equations. As an arbitrary Lagrangian–Eulerian [3] formulation of Navier–Stokes equations is generally used, residuals also depend on mesh deformation speed, $\mathrm{d}x^m/\mathrm{d}t$.

2) Eq. (2) is a general nonlinear form of structure equations. For a linear structural model this equation is usually written

$$M\frac{\mathrm{d}^2 x^s}{\mathrm{d}t^2} + C\frac{\mathrm{d}x^s}{\mathrm{d}t} + Kx^s = F(W) \tag{5}$$

3) Eq. (3) represents a mesh deformation algorithm that computes coordinates of the whole fluid mesh given coordinates of the structure mesh.

4) Eq. (4) represents the forcing applied to the system, which is an input of the problem. For the sake of clarity, only a forcing applied on the structural part is considered. However, the method developed here could deal with a forcing applied to the fluid part of the system: for example, through a periodic variation of boundary conditions.

As this paper is related to aeroelasticity, only mechanical fluid–structure coupling is considered. Nevertheless, the method presented here can easily be extended to other types of fluid–structure interactions, such as thermal coupling.

B. Usual Strategies for Solving Aeroelasticity Equations

Strategies for solving the general aeroelastic problem are numerous and include the following:

1) A monolithic solver solves both structural and mechanical equations using a single numerical formulation and a single set of variables [4,5].

2) A code-coupling algorithm uses two coupled solvers: one dedicated to fluid and one dedicated to structure [6].

3) Low-computation-cost methods are based on approximate modeling of the equations [7,8].

The method proposed in the current paper belongs to the family of the code-coupling strategies. The main interest of the code-coupling strategy is the possibility to use software specifically developed for both structure and fluid. This is interesting from both an algorithmic and an industrial point of view: dedicated solvers are generally highly efficient and optimized and they can be directly taken from tools widely used separately in both fluid and structural engineering.

The main difficulty of code-coupling is to design a strategy for using and exchanging data between the two solvers. The basic procedure for this is referred to by Piperno et al. [6] as the conventional serial staggered (CSS) algorithm. This procedure reads as follows:

1) Deform the fluid mesh according to the structure position.

2) Advance the fluid system and compute stresses acting on the structure.

3) Advance the structural system: compute displacements induced by fluid stresses.

4) Return to step 1.

The basic CSS suffers from two main issues:

1) It has poor accuracy, usually one order lower than that of the time-marching scheme used for the fluid and/or structure solver.

2) The fluid and structure equations are not computed exactly at the same physical time, and so the equilibrium between the two systems is not exactly satisfied. Piperno et al. [6,9] show how this time staggering adds some spurious energy at the interface, leading to erroneous prediction of stability properties of the system.

Numerous improvements of the CSS have been proposed to tackle these issues. They consist in using subcycling or predictor-corrector steps to reduce the staggering between the two solvers [10,11]. With these corrections the code-coupling strategy gives good results and it is currently used widely in research and industry [12,13]. Nevertheless, even with these improvements, the code-coupling strategy requires the use of small time steps to ensure the stability and accuracy of the computation. This results in high computation times.

The method presented in this paper is a new code-coupling strategy that aims at reducing the computation time of an aeroelastic system response to a forced periodic excitation. It is based on the assumption that the response of the system to a periodic excitation will generally also be periodic. Thus, it is proposed to adapt the mathematic analysis, based on this time-periodicity, that has been used for deriving the THB formulation for pure-CFD solvers for deriving the new THB-based code-coupling procedure.

III. Code-Coupling Based on the Time-Harmonic-Balance Technique

A. Time-Harmonic-Balance Technique in CFD

THB is a time integration method that has been developed in recent years, mainly at Stanford University and Duke University [14,15], for fluid dynamics computations. This method aims at reducing the computation costs for computing time-periodic flows by using a harmonic technique adapted for time-domain CFD codes.

THB can be seen as an adaptation of the classical harmonic balance (HB) technique proposed by Krylov and Bogoliuboff in 1947 [16] for computing periodic responses of nonlinear systems. The underlying idea of HB is to solve a given differential equation in the frequency domain, rather than in the time domain, by using the Fourier transformation of the equation. Solving the equation in the frequency domain leads to a system of equations (two per harmonics and one for the mean) that are solved simultaneously. When the equations for each harmonics are solved, the harmonics are said to be *balanced*, giving its name to the method. The main advantage of solving the equations in the frequency domain is to ensure that the solution obtained will be truly periodic.

The main bottleneck of this method is that the equations of the problem considered have to be transcribed and solved in the frequency domain, which can be a tedious task for complex systems of equations such as Navier–Stokes. Note, however, that this technique is still successfully used for solving aeroelasticity problems by applying it to simplified equations [17].

This bottleneck was certainly the main motivation of the work that led to the development of a version of the HB technique that can be used to solve equations written in the time domain. The first results with this technique were obtained by Hall et al. [14,18]. The method they developed can work with the semidiscretized Navier–Stokes equations implemented in a usual time-domain flow solver. The simultaneous and coupled resolution of the equations for each harmonics is replaced by its equivalent in the time domain: a coupled computation of several snapshots of the flow taken at several time steps done by several usual time-domain flow solvers running in parallel. The main difficulty with this technique is the coupling of the parallel CFD computations that requires switching between the frequency and time domains. However, this technique well achieved its main goal: directly computing periodic states of flows without resolving transients using time-domain solvers.

Gopinath and Jameson [15] and van der Weide et al. [19] proposed an extension of the method of Hall et al. that greatly simplified the coupling of flow solvers running in parallel. The coupling is reduced to the addition of a simple source term computed with an explicit formula that involves each of the CFD computations running in parallel [15,19]. Although they named this method TSM (time spectral method), the name THB will be used in this paper, following Dimitriadis [20], who places TSM in the family of HB-based methods used in aeroelasticity. THB applied to pure CFD is briefly presented here, following Gopinath [21].

Consider Navier–Stokes equations

$$\frac{\partial W}{\partial t} = R(W) \tag{6}$$

As this equation is satisfied at any time, we consider that W is an array containing the values of the conservative variables for a set of equally spaced time steps in the flow period. Moreover, for clarity, it is supposed that time has been scaled so that the flow period is equal to unity:

$$W = [w_0, w_1, \ldots, w_{N_t-2}, w_{N_t-1}]^T$$

As shown by Gopinath [21] N_t has to be taken odd. Supposing that the flow is periodic in time, the time series W can be approximated by its inverse discrete Fourier transform (IDFT). The precision of the approximation depends on the number of harmonics used for the computation. The relation linking the number of time steps to the number of harmonics computed is given by Fourier analysis theory, $N_t = 2N_h + 1$:

$$\forall\ k \in [0, \ldots, N_t - 1] w_k = \sum_{n=-\frac{N_t}{2}}^{\frac{N_t}{2}-1} \hat{w}_n e^{\frac{jkn2\pi}{N_t}} \tag{7}$$

In the following, the subscript k will be used to identify the time steps. Using this expression of w_k has two advantages:

1) It enforces the periodicity of the computed solution.

2) It eases the time derivation of W.

Using this expression, the time derivative is written:

$$\forall\ k \in [0, \ldots, N_t - 1] \frac{\partial w_k}{\partial t} = \sum_{n=-\frac{N_t}{2}}^{\frac{N_t}{2}-1} 2\pi jk \hat{w}_n e^{\frac{jkn2\pi}{N_t}} \tag{8}$$

In the above expression, the coefficient of the IDFT, $\hat{w}_n$, can be replaced by its expression as a function of the time values of W, w_k, given by discrete Fourier transform,

$$\forall\ n \in \left[1, \ldots, \frac{N_t - 1}{2}\right] \hat{w}_n = \frac{1}{N_t} \sum_{k=0}^{N_t-1} w_k e^{\frac{-jk2\pi n}{N_t}} \tag{9}$$

leading to an expression of the time derivative operator that 1) enforces the periodicity of W, 2) depends only on the time values of W, and 3) is generally of higher precision than a numerical approximation based on finite differences:

$$\forall\ k \in [0, \ldots, N_t - 1] \frac{\partial w_k}{\partial t} = \sum_{m=\frac{1-N_t}{2}}^{\frac{N_t-1}{2}} d_m w_{k+m} = D_k \cdot W \tag{10}$$

where

$$d_m = \begin{cases} \frac{\pi(-1)^{m+1}}{\sin(\frac{m\pi}{N_t})} & \text{if } m \neq 0 \\ 0 & \text{if } m = 0 \end{cases} \tag{11}$$

and D_k is a vector assembled to write the sum of Eq. (10) as a dot product that represents a time-harmonic derivation operator. In the following, the matrix $D = [D_0, \ldots, D_{N_t-1}]$ will be used. D satisfies the following relation:

$$\forall\ k \in [0, \ldots, N_t - 1] D_k \cdot W = (D \cdot W)_k \tag{12}$$

The semidiscretized Navier–Stokes equation can then be written using the above expression of the time derivative of W:

$$\forall\ k \in [0, \ldots, N_t - 1] (D \cdot W)_k = R(w_k) \tag{13}$$

This steady equation is then solved by a pseudo-time-stepping technique using the following procedure:

1) For each time step, compute $R(w_k)$ using a usual time-domain CFD solver.

2) Compute the source terms $D_k \cdot W$ that couple all time steps.

3) For each time step, update w_k and go back to step 1 until convergence has been reached.

Note that only step 2 requires a coupling of computations corresponding to each time step; steps 1 and 3 can be done independently in parallel. Moreover, as only steady Navier–Stokes equations (to which the THB source term has been added) are solved, all the advanced numerical techniques developed for such problems can be used.

The pure CFD-THB has proved its efficiency on numerous computational cases, especially for turbomachinery flows [19,22,23]. In the field of aeroelasticity, it has been used to accelerate the computation of periodic flow induced by a forced motion of a body in order to derive aerodynamic forces model required by flutter analysis methods. But to the knowledge of the authors, no solution has been proposed for using THB to solve aeroelastic problems modeled by a time-domain coupling between a CFD and a structural mechanics solver. The closest to state of the art are the works of Thomas et al. [24,25], who proposed a procedure for computing limit cycle oscillations (LCOs) using a harmonic-balance-based CFD solver coupled to structural model. This procedure differs from the coupling strategy presented here, because the coupling is done in the frequency domain, considering only the first harmonic of the aerodynamic forces. To that extent, the THB-based code-coupling procedure presented in this paper is new.

B. Extension of THB to the Equation of Aeroelasticity

Considering that a periodic phenomenon is sought, the aeroelasticity equations can be written for each element of a set of N_t time steps equally spaced in the period.

For all $k \in [0, \ldots, N_t - 1]$,

$$\frac{\partial w_k}{\partial t} = R\left(w_k, \frac{\mathrm{d}x_k^m}{\mathrm{d}t}, x_k^m\right) \tag{14}$$

$$\text{Struct}\left(\frac{\mathrm{d}^2 x_k^s}{\mathrm{d}t^2}, \frac{\mathrm{d}x_k^s}{\mathrm{d}t}, x_k^s, F(w_k)\right) = 0 \tag{15}$$

$$x_k^m = \text{MeshDef}(x_k^s) \tag{16}$$

$$\exists i \text{ such that: } x_k^s[i] = g_T(t_k) \tag{17}$$

The THB procedure is then applied to these equations, leading to approximating the time derivative using the time-harmonic derivation operator. This leads to a system of coupled static aeroelastic problems:

For all $k \in [0, \ldots, N_t - 1]$,

$$(D \cdot W)_k = R(w_k, (D \cdot x^m)_k, x_k^m) \tag{18}$$

$$\text{Struct}\,((D^2 \cdot x^s)_k, (D \cdot x^s)_k, x_k^s, F(w_k)) = 0 \tag{19}$$

$$x_k^m = \text{MeshDef}(x_k^s) \tag{20}$$

$$\exists\ i \text{ such that: } x_k^s[i] = g_T(t_k) \tag{21}$$

To solve these equations, a strategy has been derived from the fixed-point algorithm usually used for solving static aeroelasticity problems [12]:

1) Solve Eq. (18) with the usual CFD-THB procedure described in Sec. III.A.

2) Compute $F(w_k)$, the forces applied by the fluid on the structure at each time step.

3) Solve the steady equations of the structure [Eq. (19)].

4) Underrelax the displacements computed:

$$x_n^s = \breve{x}_n^s + \alpha_r (x_n^s - \breve{x}_n^s) \tag{22}$$

where the subscript n is the iteration number, $\breve{x}_n^s$ is the previous computed displacement, and α_r is the relaxation parameter used to ensure the convergence of the whole algorithm [26]. This underrelaxation scheme is a general method used in fixed-point algorithms to ensure the convergence of these algorithms. The simplest solution is to take the relaxation parameter constant for the whole computation. On some physical problems, better results are obtained by recomputing the relaxation parameter for each iteration, which can be done with a generic procedure such as Aitken method [27] or with a more advanced computation: for example, involving Steklov–Poincaré operators of fluid and structure [28,29]. In the field of aeroelasticity, using a constant relaxation coefficient is sufficient, since added mass effects are generally negligible [10]; thus, a constant coefficient has been used in the present study.

5) Apply the displacement of the structure on the fluid mesh [Eq. (20)]. This displacement is applied by transferring the displacement form the structural mesh to the skin of the body in the fluid mesh and then by transferring these displacements form the skin to the whole volume. The structural displacements include all the structure degrees of freedom plus the displacements corresponding to the forced motion.

6) Return to step 1 until convergence has been reached.

As with the CFD problem, most of the computations for solving the different static aeroelastic problems can be done separately. The coupling of each computation is done during step 1 to solve the fluid equations (as explained in Sec. III.A) and structure equations (step 3).

Two practical means can be used to save some computation time:

1) As CFD computations are usually the most expensive part of the whole computation, CFD processes should not be run from scratch at each iteration of the fluid–structure process: it is better to use the flow solution obtained at the previous iteration to initialize the CFD computation.

2) Using the above, it might not be necessary to wait for full convergence of the fluid solver at each iteration of the process: time can be saved by iterating in the CFD solver less than required for convergence, with the convergence of the fluid solver being reached with the convergence of the whole process.

One important feature of the THB-based coupling algorithm is that fluid and structure are not staggered in time: the equilibrium of the two systems does not rely on the accuracy of a predictor step [30]. This should lead to a reduced production of spurious energy at the fluid–structure interface, at least when the convergence of the algorithm has been reached.

The THB-based code-coupling strategy has basically the same limitations as THB of CFD:

1) With THB, the frequency used for performing the harmonic analysis is an input of the method, and therefore the frequency of the phenomena must be known before the computations. For this reason, the THB-based code-coupling algorithm cannot be used for computing self-determined aeroelastic phenomena such as LCOs occurring at a frequency unknown before the computation. In the case of vibration cycles caused by a forced motion, the frequency of the vibrations will generally be the same as that of forced motion. The frequency of the forced motion can then be used directly for performing the THB. Some extensions of pure CFD-THB extensions, based on iterative procedures for finding the unknown frequency, have been created [31,32]. Thomas et al. [24,25] also proposed a procedure for computing LCOs occurring at an unknown frequency using a HB-based CFD solver and a structural model. Extending these methods to the THB-based code-coupling algorithm could be a way for improvement.

2) As THB works with a single frequency, the computation of phenomena with several different frequencies is not possible. Thus, the THB-based coupling algorithm cannot be used for computing the vibration cycles of an aeroelastic system with several forcings done at different frequencies (for example, a wing with two ailerons oscillating at different frequencies). Also in this field, THB improvements are available to compute flows varying with several frequencies [33]. These could be used to remove this limitation of the THB-based code-coupling algorithm.

The clearest view of the coupling procedure can be obtained by considering a linear structure whose equations are written in an eigenvector basis. In this basis, structure equations are a set of uncoupled scalar ordinary differential equations. The unknowns are the generalized displacements along each eigenmode. The eigenvector basis used is usually truncated, and only a limited number of eigenmodes, associated with the lowest eigenfrequencies, are used:

$$\forall\ i \in [1, \ldots, N_s - 1]\colon m_i \frac{\mathrm{d}^2 x^s[i]}{\mathrm{d}t^2} + c_i \frac{\mathrm{d}x^s[i]}{\mathrm{d}t} + k_i x^s[i] = f_i(W) \tag{23}$$

$$i = N_s\colon x^s[i] = g_T(t) \tag{24}$$

In the following, the subscript k will be used to identify the structural modes, and x^s corresponds to the vector of generalized displacements; chosen coordinates are the generalized displacements along each eigenmode.

Introducing the time-harmonic derivation operator in Eq. (23) leads to

$$\forall\ (i,k) \in [1, \ldots, N_s - 1] \times [0, \ldots, N_t - 1]\colon m_i (D^2 x^s[i])_k + c_i (D x^s[i])_k + k_i x_k^s[i] = f_i(W) \tag{25}$$

$$(i,k) = N_s \times [0, \ldots, N_t - 1]\colon x_k^s[i] = g_T(t_k) \tag{26}$$

which can be written

$$\forall\ i \in [1, \ldots, N_s - 1]\colon \Gamma_i x[i] = f_i(W) \tag{27}$$

$$(i,k) = N_s \times [0, \ldots, N_t - 1]\colon x_k^s[i] = g_T(t_k) \tag{28}$$

where Γ_i is the matrix associated to the ith mode, resulting from the time-discretization of Eq. (23) using the time-harmonic derivation operator D:

$$\Gamma_i = m_i D^2 + c_i D + k_i I \tag{29}$$

With such a simplified structural model, the structural problem can be solved simply by computing the result of the linear problem (27).

IV. Numerical Results

Two test cases are presented in this section: the first one aims at analyzing and validating the results given by the THB-based code-coupling strategy on a simple academic case. The second one is used to test the method on a larger, industrial-type, test case.

A. Two-Dimensional Pitch-Plunge NACA64A010 Airfoil

This test case consists of a two-dimensional NACA64A010 airfoil mounted on plunge spring excited by a forced pitching motion (see Fig. 1). The computation aims at finding the periodic plunge vibration caused by the forced pitching motion. The main physical parameters of the computation are summarized in Table 1. The equation used for the forced motion is $\theta = \theta_m \sin(\omega_m t)$. A simple linear spring-mass model was used for the structure model that is reduced to its plunging degree of freedom:

$$m\frac{\mathrm{d}^2x^s[i]}{\mathrm{d}t^2} + c\frac{\mathrm{d}x^s[i]}{\mathrm{d}t} + kx^s[i] = f(W)$$

For the fluid part, the unsteady Reynolds-averaged Navier–Stokes equation with the Spalart–Allmaras [34] turbulence model were solved using Jameson's scheme [35], with a LU-SSOR (lower/upper symmetric successive overrelaxation) implicit phase [36]. For the implicit phase, the specific treatment of the THB source term proposed by Sicot et al. [37] has been used. A multigrid technique, with two coarse grids, was used to accelerate the convergence. All fluid dynamics computations were done using the elsA [38] flow solver. The mesh deformation algorithm used is based on the transfinite interpolation algorithm [39]. The fluid mesh has around 80,000 cells. The mesh and the numerical parameter tuning for this case have been done following previous works with classical solvers [12]. All computations were done in parallel on two processors.

1. *Computation Using Standard Coupling Algorithm*

These computations were done using the fluid–structure coupling capabilities available in elsA [13]. The time advancement of the fluid was done using a dual-time-stepping scheme [40]. The number of physical time steps was set to 40 physical time steps per period of the forced motion, and subiterations were stopped when 2 orders of magnitude were lost on density residuals. The structure dynamics is advanced in time using Newmark's scheme [41].

The coupling between fluid and structure is done using a code-coupling algorithm based on the CSS, with subcycles between fluid and structure to ensure the equilibrium between fluid and structure at each time step [13]. Two subiterations of fluid–structure coupling were done per time step. To accelerate the convergence of the computation toward a periodic solution, some damping was added to the structure to dissipate the transients. This damping decreases during the computation following the formula $c(t) = 2\sqrt{2mk}e^{-(\frac{\omega_F t}{2\pi})^2}$, which ensures that this damping is negligible after four periods of forced motion.

Results are presented in Fig. 2 in a position-velocity diagram of the airfoil position along its plunge axis. This was obtained after 25 periods of forced motion. On this diagram, the trajectory that corresponds to a periodic motion is a closed shape. It appears that the transient is very long and that even after 25 periods of forced motion the periodic state is not completely reached, as the trajectories corresponding to the last two periods are not identical.

The result presented in Fig. 2 required around 64,000 pseudo time iterations of the flow solver and 1000 structural computations.

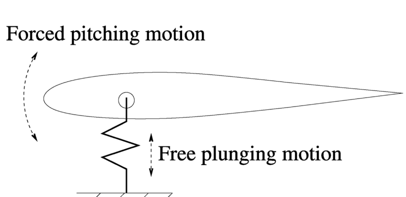


Fig. 1 NACA64A010 free-plunge/forced-pitch test case.

Table 1 Physical parameters for the NACA64A010 airfoil

Parameter	Value
k	$30.0\ \mathrm{N\cdot m^{-1}}$
c	$0.0\ \mathrm{N\cdot m^{-1}\cdot s}$
m	3.0 kg
ω_m	$3.89\omega_{\mathrm{struct}}$
Airfoil chord	1 m
Mach number	0.796
Reynolds number	13×10^6
θ_m	1.01 deg

2. *Computation Using THB-Based Coupling Algorithm*

The THB-based coupling strategy developed in Sec. III.B was applied to the airfoil test case. As explained, the THB-coupling algorithm transforms the dynamic aeroelasticity problem into several coupled static aeroelasticity problems. Figure 3 shows a sketch representing the equivalent static problems solved for $N_t = 3$. It has been supposed that the frequency of the response is equal to the frequency of the forcing applied by the pitching motion.

As proposed in Sec. III.B, the CFD computations were not fully converged at each iteration of the code-coupling scheme, and the results of the CFD computations corresponding to previous time steps were reused to initialize the CFD computation at the following iteration. The structural equations are solved using a simple Gauss elimination procedure. At each iteration of the coupling scheme, 75 pseudo time iterations of the CFD solver were done. The under-relaxation parameter α_r was set to 0.5.

Figure 4 presents the computed position-velocity diagram of the airfoil along the plunge axis, for $N_t = 5$ (corresponding to two harmonics plus the mean). Each ellipsoid represents the computed vibration cycle at each iteration of the fluid–structure coupling scheme. The algorithm clearly appears to converge toward a unique and stabilized periodic-vibration cycle. These results were obtained after 1000 fluid solver iterations, which corresponds to 13 iterations of the coupling algorithm (thus, 13 resolutions of the structure equations).

Note that the THB-based coupling algorithm computes only some points of the vibration cycles. Each of them is represented by the triangles in Fig. 4. These computed points correspond to the result of each of the coupled static aeroelasticity problems used by the process. The curves presented in Fig. 4 were obtained using a harmonic interpolation between the computed points.

Figures 5 and 6 show vibration cycles and the unsteady pressure coefficient computed with different values of N_t (unsteady pressure coefficients on the airfoil are represented by their mean value plus the real and imaginary parts of the first harmonics). The agreement between the results for each value is satisfactory, and a convergence is observed when N_t is increased.

3. *Results Comparison*

Figures 7 and 8 compare the vibration cycles and the unsteady pressure coefficients computed with the standard code-coupling

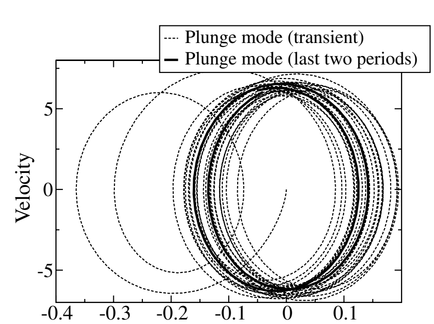


Fig. 2 Position-velocity diagram of the airfoil position along the plunge axis.

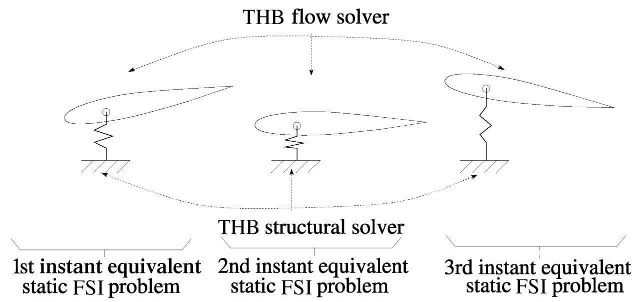


Fig. 3 View of the coupled static aeroelasticity problems solved by the THB-based coupling algorithm for $N_t = 3$ (FSI denotes fluid–structure interaction).

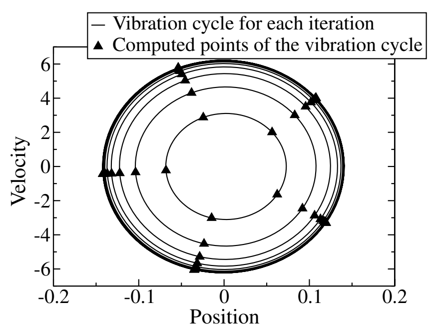


Fig. 4 Convergence of the THB-based coupling strategy: view of computed vibration cycles for each iteration of the code-coupling process. $N_t = 5$.

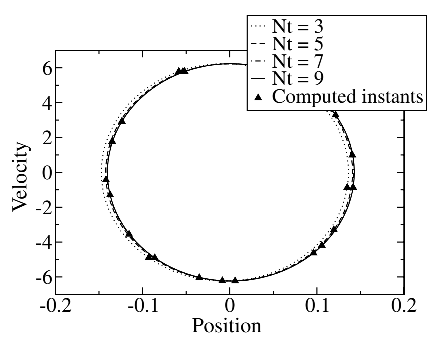


Fig. 5 Convergence of the THB-based coupling strategy: view of computed vibration cycles for several values of N_t.

algorithm and the THB-based coupling. The agreement of the results of the two methods is satisfactory.

For this case, with the use of the THB-based coupling algorithm, the number of CFD solver iterations required to complete the computation was divided by 64 and the number of resolutions of structural equations was divided by 75. However, it has to be considered that each iteration of the THB-based coupling algorithm is more expensive, both in time and memory, than an iteration of the standard code-coupling algorithm. This is because the THB-coupling algorithm requires simultaneously solving several coupled steady problems. For example, with $N_t = 5$, five coupled steady problems are solved by the THB-based coupling; thus, the cost of one iteration of the THB-based algorithm is roughly five times the cost of one iteration of the time-domain coupling algorithm.

Table 2 summarizes the computation times and memory usage observed for computations with the standard code-coupling algorithm and the THB code-coupling with several values of N_t. All the computations were done on two AMD-Opteron processors. It appears in this table that despite the increased cost of each iteration, computation with the THB-based coupling algorithm is significantly

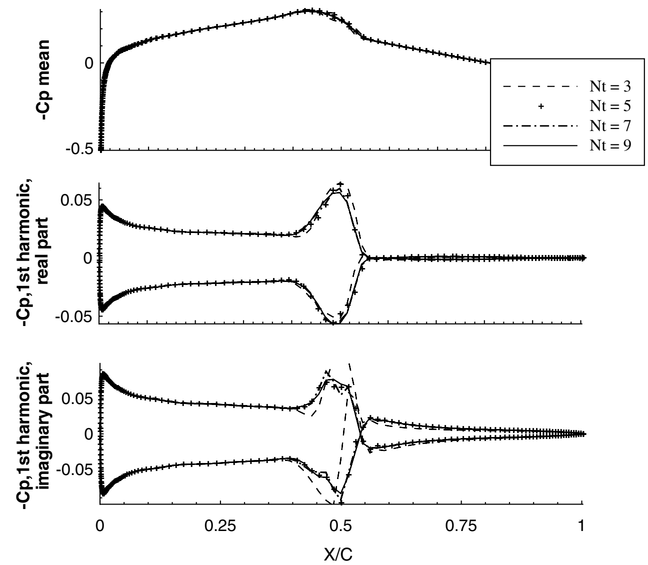


Fig. 6 Convergence of the THB-based coupling strategy: view of computed unsteady pressure coefficients for several values of N_t.

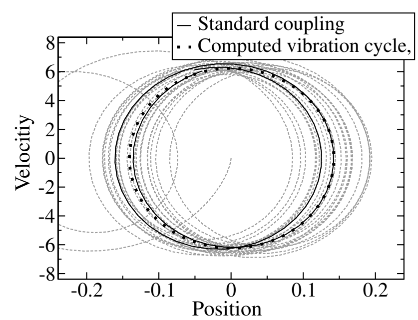


Fig. 7 Comparison of the vibration cycles computed with the standard code-coupling algorithm and with the THB-based coupling ($N_t = 5$).

faster than computation with the standard code-coupling algorithm. The increase of memory usage, due to the use of several coupled steady problems by the THB-based algorithm, can also be seen in this table.

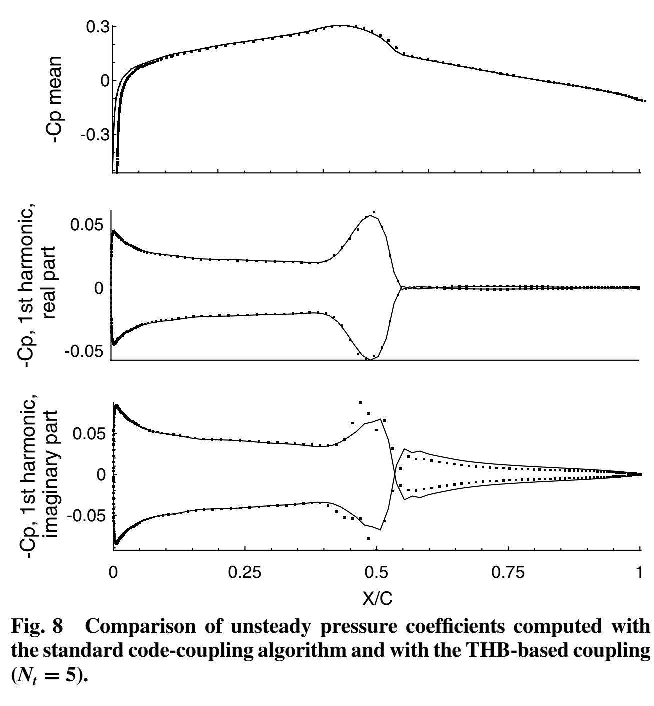


Fig. 8 Comparison of unsteady pressure coefficients computed with the standard code-coupling algorithm and with the THB-based coupling ($N_t = 5$).

Table 2 Computation characteristics

Computation	Computation time	Required memory
Standard code coupling	16 h, 15 min	0.32 GB
THB code coupling, $N_t = 3$	1 h, 30 min	0.91 GB
THB code coupling, $N_t = 5$	2 h, 40 min	1.51 GB
THB code coupling, $N_t = 7$	3 h, 50 min	2.10 GB
THB code coupling, $N_t = 9$	4 h, 45 min	2.69 GB

B. Unsteady Flexibility Effect on a Wind-Tunnel Model

In this section the THB-based coupling algorithm is applied to an industrial study to take into account the flexibility of a wind-tunnel model in the computation of unsteady aerodynamic coefficients on a wing with a moving aileron. The CFD mesh used for the resolution of RANS equations has 3 million nodes (see Fig. 9). The Reynolds and Mach numbers of the flow are, respectively, equal to 6.46×10^5 m^{-1} and 0.89. A reduced structural model limited to the 10 first eigenmodes of the full structural model is used (see Fig. 10). The ratio between the highest and the lowest eigenfrequencies kept is around 27. The structural model does not contain any damping. The amplitude of the aileron motion is 6 deg and its frequency corresponds to the frequency of the first eigenmode minus 6%. It has been supposed that the frequency of the response is equal to the frequency of the forcing applied by the aileron motion.

The computation was done with $N_t = 5$. Before the initial fluid–structure coupling, 200 iterations of the CFD solver were done to initialize the flow solution. The initial state of the structural equations was simply taken equal to zero for both position and velocity of each eigenmode. After the first coupling step, the coupling was done every 100 iterations of the fluid solver. The relaxation parameter was taken equal to $\alpha_r = 0.5$. Thirteen couplings between fluid and structure were done. The whole computation required 1500 iterations of the flow solver and 13 structural computations.

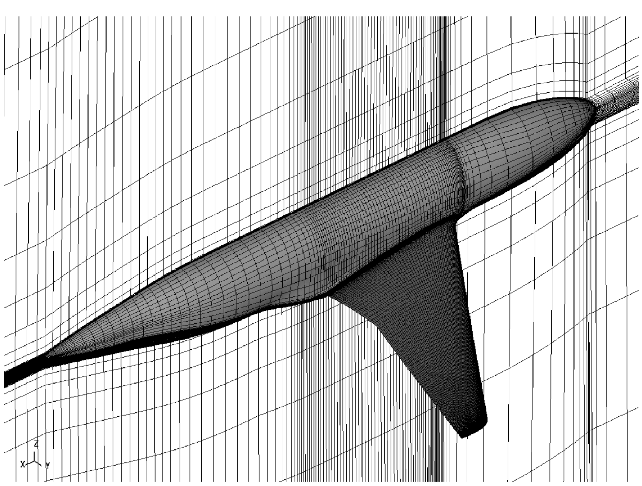


Fig. 9 View of the mesh used for CFD computations.

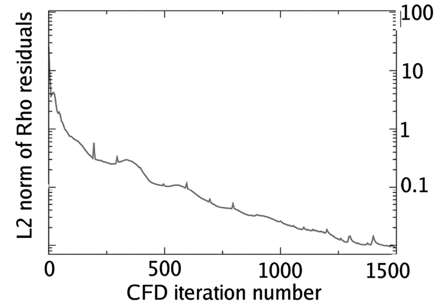


Fig. 11 Concatenated ρ residuals of all CFD computations ran.

Figure 11 plots the concatenated ρ residual curves of all CFD computations run by the coupling algorithm. As explained in Sec. III.B, each CFD computation was started using the solution obtained at the previous coupling step as an initial step. On this figure the small jumps on the curve correspond to the restarts of the CFD computation after each structure computation. As the overall residual curve decreases from one coupling step to the next, it can be stated that the overall coupling process converges.

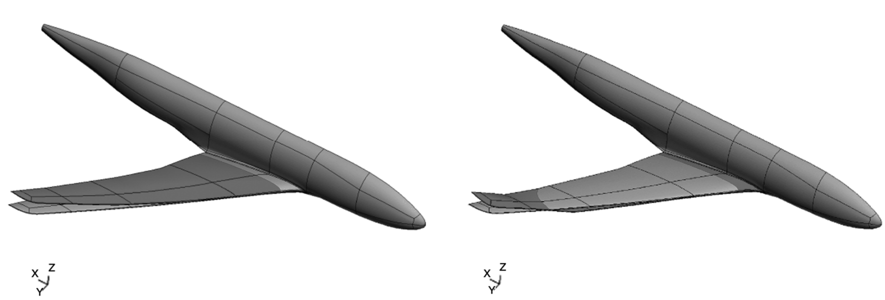
From a structural point of view, the convergence of the coupling process can be seen in Fig. 12 that presents the computed vibration cycle for each structural mode and for each iteration of the structural solver. It can be seen that for each structural mode, the computed vibration cycle converges toward a stabilized cycle. However, it also appears that the convergence of the vibration cycles is slightly slower on the last coupling steps. This might be caused by the use of a fixed value of the underrelaxation parameter used. Recomputing this relaxation parameter using, for example, the Aitken procedure would be a way to improve the convergence rate on the last iterations.

Figure 12 also shows that during the computation the mean position of each structural mode moves during the computation. This is because the initial coordinates of the structural system do not correspond to those given by the mean aerodynamic forces generated at the flight conditions used for the computation.

The computation took around 10 h on eight PowerPC Power5 (dual-core) processors, which is an appropriate value for an industrial use of the method, since computations can be run in a night. Unsurprisingly, the global memory usage is around 26 GB: that is, higher than the memory usage of a computation with a time-marching scheme, which would be around 6 GB for this case. On computers with only a small memory per processor available, it could be necessary to increase the number of processors used for the computation.

V. Conclusions

The time-harmonic-balance method initially developed for CFD has been applied to computation of the periodic-vibration cycles of aeroelastic systems. The solver obtained can directly compute a periodic-vibration cycle without computing the long physical transients, leading to significant reduction of the computational effort required. The most interesting point with this THB-based coupling-procedure is certainly that it can be applied to any standard time-domain code-coupling procedure without transformation to the



a) Mode 0 b) Mode 1

Fig. 10 View of the first two eigenmodes used for the structural model (displacements magnified 10 times).

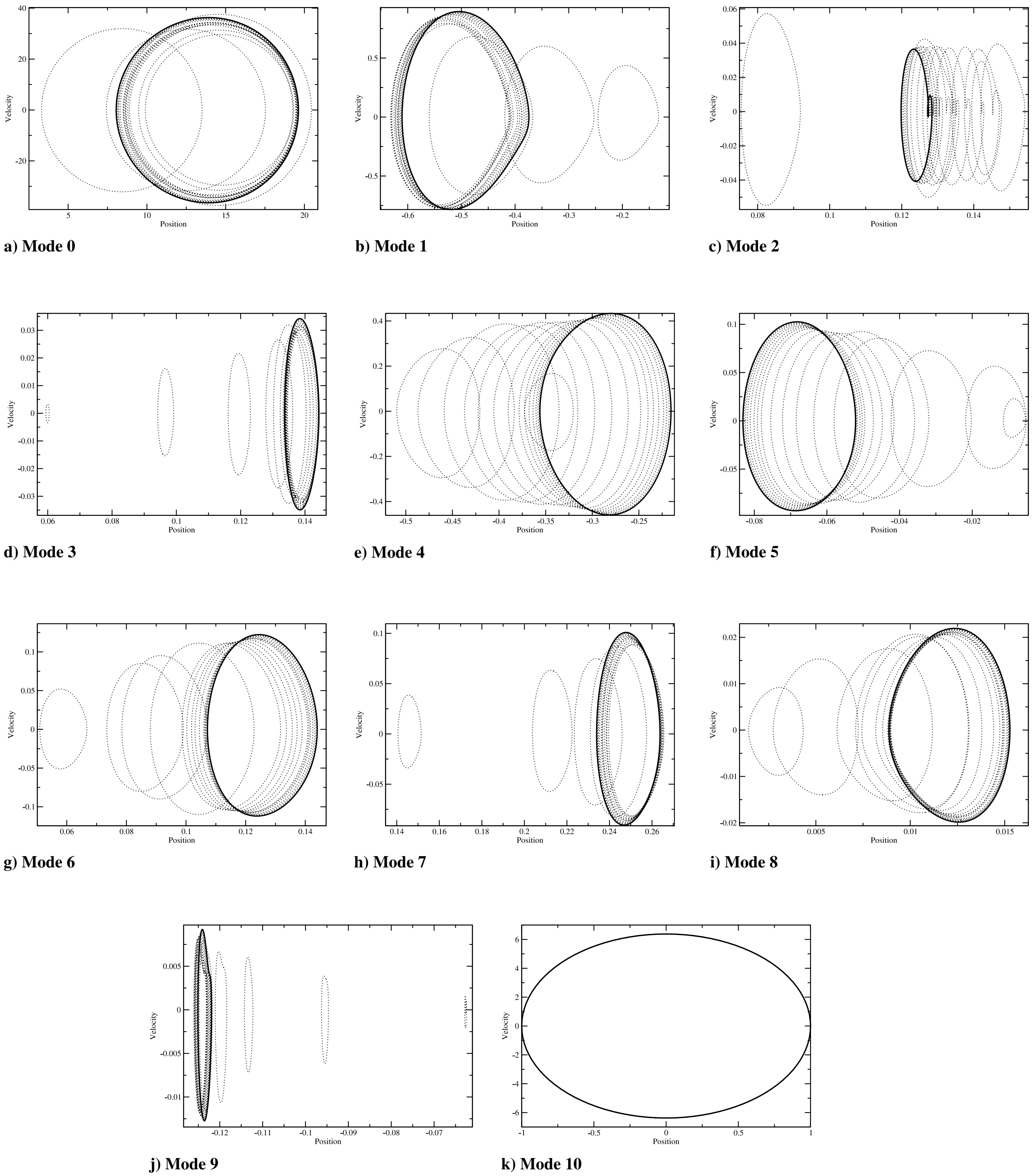


Fig. 12 Computed position-velocity diagram for each structural mode; plain: final computed vibration cycles, dotted: transient vibration cycles obtained at each iteration of the THB-based code-coupling procedure. Mode 10 corresponds to the aileron forced motion.

frequency domain. Lacking any experimental data, the validation of the THB-based code-coupling procedure has been done only by comparison with the usual time-marching aeroelastic solver. The results given by the proposed solver are in line with results obtained with the usual aeroelastic solver and it has been shown that the method is able to process large industrial configurations. However, the validation of the results by comparison to some experimental data remains to be done.

Because of the low computational cost of the THB-based coupling strategy, it is possible to take flexibility effects into account in the computation of unsteady aerodynamic coefficients.

Further works focus on adding the capability for the solver to compute vibration cycles induced by forcings applied with several different frequencies and to compute self-induced limit cycle oscillations occurring at frequencies unknown before the computation.

Acknowledgments

This work is part of thesis research funded by Airbus and conducted in collaboration with Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique (CERFACS). The authors wish to thank all the people of the Airbus Aerodynamic Methods and

Tools team in Toulouse. The elsA developers from CERFACS and ONERA Aeroelasticity and Structural Dynamics unit, and especially Guillaume Dufour and Frederic Sicot, are gratefully acknowledged for their help on the time-harmonic-balance technique and on the numerical methods for aeroelasticity.

References

[1] Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., *Aeroelasticity*, Addison-Wesley, Reading, MA, 1955.

[2] Dowell, E. H. E., *A Modern Course in Aeroelasticity*, 3rd ed., Kluwer Academic, Norwell, MA, 1995.

[3] Donea, J., Giuliani, S., and Halleux, J. P., "An Arbitrary Lagrangian Eulerian Finite Element Method for Transient Dynamic Fluid-Structure Interactions," *Computer Methods in Applied Mechanics and Engineering*, Vol. 33, 1982, pp. 689–723.
doi:10.1016/0045-7825(82)90128-1

[4] Michler, C., Hulshoff, S. J., Van Brummelen, E. H., and De Borst, R., "A Monolithic Approach to Fluid-Structure Interaction," *Computers and Fluids*, Vol. 33, 2004, pp. 839–848.
doi:10.1016/j.compfluid.2003.06.006

[5] Blom, F., "A Monolothical Fluid-Structure Interaction Algorithm Applied to the Piston Problem," *Computer Methods in Applied Mechanics and Engineering*, Vol. 167, 1998, pp. 369–391.
doi:10.1016/S0045-7825(98)00151-0

[6] Piperno, S., Fahrat, C., and Larrouturou, B., "Partitioned Procedures for the Transient Solution of Coupled Aeroelastic Problems. Part I : Model Problem, Theory and Two-Dimensional Application," *Computer Methods in Applied Mechanics and Engineering*, Vol. 124, 1995, pp. 79–112.
doi:10.1016/0045-7825(95)92707-9

[7] Irwin, C. A., and Guyett, P. R., "The Subcritical Response and Flutter of a Swept Wing Model," Royal Aircraft Establishment, TR 3497, London, 1965.

[8] Chen, P. C., "Damping Perturbation Method For Flutter Solution : the g-method," *AIAA Journal*, Vol. 38, 2000, pp. 1519–1524.
doi:10.2514/2.1171

[9] Piperno, S., and Farhat, C., "Partitioned Procedures for the Transient Solution of Coupled Aeroelastic Problems. Part II : Energy Transfer Analysis and Three-Dimensional Applications," *Computer Methods in Applied Mechanics and Engineering*, Vol. 190, 2001, pp. 3147–3170.
doi:10.1016/S0045-7825(00)00386-8

[10] Piperno, S., "Explicit/Implicit Fluid/Structure Staggered Procedures with a Structural Predictor and Fluid Subcycling for 2D Inviscid Aeroelastic Simulations," *International Journal for Numerical Methods in Fluids*, Vol. 25, 1997, pp. 1207–1226.
doi:10.1002/(SICI)1097-0363(19971130)25:10<1207::AID-FLD616>3.0.CO;2-R

[11] Geuzaine, P., van der Zee, K., and Fahrat, C., "Second Order Time Accurate Loosely Coupled Solution Algorithm for Nonlinear FSI problems," *Proceedings of ECCOMAS 2004*, Jyvaskyla, Finland, 2004.

[12] Delbove, J., "Contribution aux Outils de Simulation Aéroélastique des Aéronefs : Prédiction du Flottement et Déformation Statique des Voilures," Ph.D. Thesis, École Nationale Supérieure de l'Aéronautique et de l'Espace, Toulouse, France, 2005.

[13] Girodroux-Lavigne, P., and Dugeai, A., "Fluid-Structure Coupling in elsA : Status of Developments and Examples of Application," *42ème Colloque d'Aérodynamique Appliquée*, Nice, France, 2007.

[14] Hall, K. C., Thomas, J. P., and Clark, W. S., "Computation of Unsteady Nonlinear Flows in Cascades Using a Harmonic Balance Technique," *AIAA Journal*, Vol. 40, No. 5, 2002, pp. 879–886.
doi:10.2514/2.1754

[15] Gopinath, A., and Jameson, A., "Time Spectral Method for Periodic Unsteady Computations over Two- and Three-Dimensional Bodies," 43rd Aerospace Sciences Meeting and Exhibit, AIAA Paper 2005-1220, Reno, NV, Jan. 2005.

[16] Krylov, N., and Bogoliuboff, N., *Introduction to Non-Linear Mechanics*, Princeton Univ. Press, Princeton, NJ, 1947.

[17] Dimitriadis, G., Vio, G. A., and Cooper, J. E., "Application of Higher-Order Harmonic Balance to Non-Linear Aeroelastic Systems," 47th AIAA Structures, Structural Dynamics and Materials Conference, AIAA Paper 2006-2023, Newport, RI, 2006.

[18] Hall, K. C., Thomas, J. P., Ekici, K., and Voytovich, D. M., "Frequency Domain Techniques for Complex and Nonlinear Flows in Turbomachinery," 33rd AIAA Fluid Dynamics Conference and Exhibit, AIAA Paper 2003-3998, Orlando, FL, June 2003.

[19] van der Weide, E., Gopinath, A., and Jameson, A., "Turbomachinery Applications with the Time Spectral Method," 35th AIAA Fluid Dynamics Conference and Exhibit, AIAA Paper 2005-4905, Toronto, June 2005.

[20] Dimitriadis, G., "Continuation of Harmonic Balance Solution for Nonlinear Aeroelastic Systems," *Journal of Aircraft*, Vol. 45, 2008, pp. 523–237.
doi:10.2514/1.32871

[21] Gopinath, A. K., "Efficient Fourier-Based Algorithms for Time Periodic Problems," Ph.D. Thesis, Stanford Univ., Stanford, CA, 2007.

[22] Hall, K. C., and Pratt, E. T., "Modeling of Unsteady Three-Dimensional Flows in Multistage Machines," NASA Tech. Rep. NASA-CR-2003-212101, Jan. 2003.

[23] Ekici, K., and Hall, K. C., "Nonlinear Analysis of Unsteady Flows in Multistage Turbomachines Using the Harmonic Balance Technique," 44th AIAA Aerospace Sciences Meeting, AIAA Paper 2006-422, Reno, NV, Jan. 2006.

[24] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Nonlinear Inviscid Aerodynamic Effects on Transonic Divergence, Flutter, and Limit-Cycle Oscillations," *AIAA Journal*, Vol. 40, No. 4, April 2002, pp. 638–646.
doi:10.2514/2.1720

[25] Thomas, J. P., Dowell, E. H., and Hall, K. C., "Modeling Limit Cycle Oscillation Behavior of the F-16 Fighter Using a Harmonic Balance Approach," 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials (SDM) Conference, AIAA Paper 2004-1696, Palm Springs, CA, April 2004.

[26] Causin, P., Gerbeau, J. F., and Nobile, F., "Added Mass Effect in the Design of Partitioned Algorithms for Fluid-Structure Problems," Institut National de Recherche en Informatique et Automatique, TR 5084, Rocquencourt, France, 2004.

[27] Irons, B. M., and Tuck, R. C., "A Version of the Aitken Accelerator for Computer Iteration," *International Journal for Numerical Methods in Engineering*, Vol. 1, No. 3, 1969, pp. 275–277.
doi:10.1002/nme.1620010306

[28] Le Tallec, P., and Mouro, J., "Fluid Structure Interaction with Large Structural Displacements," *Computer Methods in Applied Mechanics and Engineering*, Vol. 190, 2001, pp. 3039–3067.
doi:10.1016/S0045-7825(00)00381-9

[29] Deparis, S., Discacciati, M., Fourestey, G., and Quateroni, A., "Fluid-Structure Algorithms Based on Steklov–Poincare Operators," *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, 2006, pp. 5797–5812.
doi:10.1016/j.cma.2005.09.029

[30] Piperno, S., and Fahrat, C., "Design and Evaluation of Staggered Partitioned Procedures for Fluid-Structure Interaction Simulations," Institut National de Recherche en Informatique et Automatique, TR 3241, Rocquencourt, France, 1997.

[31] McMullen, M., Jameson, A., and Alonso, J., "Application of a Non-Linear Frequency Domain Solver to the Euler and Navier–Stokes Equations," 40th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2002-0120, Reno, NV, Jan. 2002.

[32] Gopinath, A., and Jameson, A., "Application of the Time Spectral Method to Periodic Unsteady Vortex Shedding," 44th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2006-0449, Reno, NV, Jan. 2006.

[33] Gopinath, A., van der Weide, E., Alonso, J., Jameson, A., Ekici, K., and Hall, K., "Three-Dimensional Unsteady Multi-Stage Turbomachinery Simulations Using the Harmonic Balance Technique," 45th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2007-0892, Reno, NV, Jan. 2007.

[34] Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Transport Model for Aerodynamic Flows," 30th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 92-0439, Reno, NV, Jan. 1992.

[35] Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA 14th Fluid and Plasma Dynamic Conference, AIAA Paper 81-1259, Palo Alto, CA, June 1981.

[36] Yoon, S., and Jameson, A., "An LU-SSOR Scheme for the Euler and Navier–Stokes Equations," AIAA 25th Aerospace Sciences Meeting, AIAA Paper 87-0600, Reno, NV, Jan. 1987.

[37] Sicot, F., Puigt, G., and Montagnac, M., "Block-Jacobi Implicit Algorithms for the Time Spectral Method," *AIAA Journal*, Vol. 46, No. 12, 2008, pp. 3080–3089.
doi:10.2514/1.36792

[38] Cambier, L., and Gazaix, M., "elsA: An Efficient Object-Oriented Solution to CFD Complexity," 40th AIAA Aerospace Science Meeting

and Exhibit, Reno, AIAA Paper 2002-0108, Jan. 2002, pp. 14–17.

[39] Tsai, H. M., Wong, A. S. F., Cai, J., Zhu, Y., and Liu, F., "Unsteady Flow Calculations with a Parallel Multiblock Moving Mesh Algorithm," *AIAA Journal*, Vol. 39, 2001, pp. 1021–1029. doi:10.2514/2.1442

[40] Jameson, A., "Time Dependent Calculations Using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings," AIAA Paper 91-1596, 1991.

[41] Newmark, N. U., "A Method of Computation for Structural Dynamics," *Journal of the Engineering Mechanics Division*, Vol. 59, No. 3, 1959, pp. 67–94.

A. Sinha
Associate Editor