[11]Elishakoff, I., and Lee, L. H. N., "On Equivalence of the Galerkin and Fourier Series Methods for One Class of Problems," *Journal of Sound and Vibration*, Vol. 119, 1986, pp. 174–177.

[12]Filonenko-Borodich, M. M., "On a Certain System of Functions and Its Applications in the Theory of Elasticity," *PMM—Prikladnaya Matematika Mekhanika*, Vol. 10, No. 2, 1946, pp. 193–208 (in Russian).

[13]Stepanov, V. V., "On Some Complete Non-Orthogonal Systems," *Proceedings of the USSR Accademy of Sciences*, Vol. 48, No. 6, 1945 (in Russian).

[14]Gradshteyn, I. S., and Ryzhik, I. M., *Tables of Integrals Series and Products*, Academic Press, New York, 1980.

S. Saigal
Associate Editor

Analysis of Time-Variant Aeroelastic Systems Using Neural Network

Anand Natarajan,* Rakesh K. Kapania,† and Daniel J. Inman‡
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0203

Introduction

THE use of smart materials[1] in wings allows active control of the deflections of the wing structure. By using adaptive material in aircraft wings, it is possible to introduce the property of time-varying stiffness. Another approach to a variable stiffness wing is through the concept of a variable stiffness spar,[2] which allows for the variation of the torsional stiffness of the wing. For a dynamical system such as an oscillating wing, the presence of a variable torsional stiffness renders the system to be time variant, that is, the parameters defining the system vary with time. This time-varying stiffness is used herein to control airfoil oscillations. It is also advantageous to have a variable stiffness wing in the case of morphing[1] wings as this allows the wing to hold the load while still being flexible enough to allow morphing of the airfoil for different flight regimes. A brief description of the present approach and results is given here. Details are available in the conference paper, AIAA Paper 2002-5599.

Aeroelastic Applications of Neural Networks

Figure 1 shows the comparison between a doublet-panel code and the theoretical Wagner solution[3] for the unsteady lift over a symmetric airfoil with a variable-amplitude pitching motion. As can be seen from the comparison made in Fig. 1, the results from an unsteady doublet-panel code developed here and from theory are very similar. In the present work, this unsteady doublet-panel method is linked to a structural dynamics solver based on the method of matrix exponential time marching.[4] Analytical solutions to dynamic aeroelastic problems are few,[3] and the presence of arbitrarily time-varying coefficients in the structural dynamic equations prevents analytical solutions. When numerical methods are used to model the dynamic aeroelastic behavior, the coupling between the aerodynamics and structural behavior can lead to very large computational costs. Most applications of neural networks in aeroelasticity have been in system parameter estimation for aircraft control or in flutter control from experimental data.[5,6] A recent work in the application of neural networks to static nonlinear aeroelastic problems in morphing wings[7] describes the procedure of aeroelastic optimization of an adaptable bump on the surface of an airfoil.

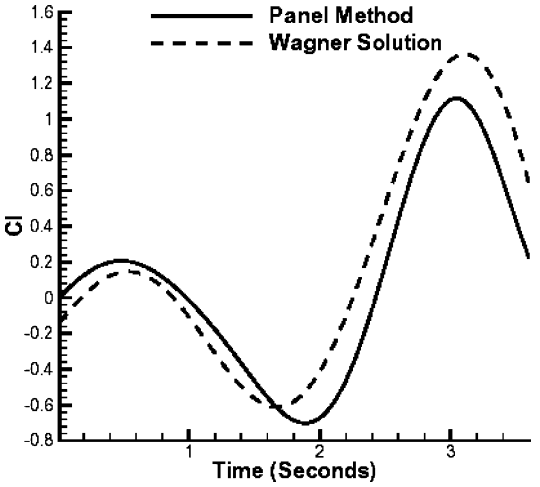


Fig. 1 Unsteady lift over a pitching airfoil with pitch angle $\alpha(t) = 0.1t \sin(2t)$ as computed by the doublet-panel code and compared with the Wagner solution.

The presence of varying coefficients in the differential equation of motion of a system significantly complicates the analysis even for a linear system. For example, traditional approaches to determine the stability, such as the Routh–Hurwitz criterion for continuous-time systems or Jury test[8] for discrete-time systems fail when the system is time varying. The second method of Liapunov, though applicable, is complicated because aeroelasticity involves nonconservative loading. Therefore herein, a stable time-domain solution for linear-time-variant systems is first explored and then utilized for training feed-forward neural networks such that a time-varying dynamic aeroelastic simulation is feasible.

Presence of noise is typical of recurrent neural networks and is one of the disadvantages of such neural networks. Hence in order to present a better match between the results as given by a recurrent neural network and those by a time-varying dynamical system, it is necessary to use smaller time steps However, use of smaller time steps increases the size of the dataset required for the network training. Large quantities of input data do not allow for accurate artificial-neural-network (ANN) training. Hence instead of using reduced time steps to make the neural-network representation of the time-varying system more accurate, it is more reasonable to study accurate numerical solvers for linear-time-varying systems that can use large time steps.

Recent advances in variable stiffness wings[2] can allow for exponentially saturating stiffness variation. Accordingly, the time-varying torsional stiffness is assumed as

$$k_\alpha(t) = k_0 + k_{01}\left(1 - c_1 e^{-c_2 t}\right) \tag{1}$$

where k_0 is the initial torsional stiffness and $k_0 + k_{01}$ is the maximum torsional stiffness. The rate of variation is controlled by the two parameters c_1 and c_2. Training of recurrent neural networks based on these numerical results for dynamic aeroelastic problems requires that the time step be small. Hence, we investigate a stable time-integration scheme for linear-time-varying systems using large time steps. The utilization of such a scheme for the training of feed-forward neural networks to represent the dynamic aeroelastic system is further investigated.

Response of Linear-Time-Varying Dynamical Systems

Linear-time-varying systems can be analyzed by the method of matrix exponential time marching.[4] Though an analytical solution

Received 26 July 2002; presented as Paper 2002-5599 at the AIAA/ISSMO 9th Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA, 4–6 September 2002; accepted for publication 7 April 2003.

*Graduate Research Assistant, Department of Aerospace and Ocean Engineering.

†Professor, Department of Aerospace and Ocean Engineering. Associate Fellow AIAA.

‡G.R. Goodson Professor, Department of Mechanical Engineering, Director, Center for Intelligent Material Systems and Structures. Fellow AIAA.

is described in Ref. 4 for single-degree-of-freedom systems, this work can be extended to multi-degree-of-freedom systems numerically. The important feature of matrix exponential time marching is that a quite large time step of the order of 10^{-2} to 10^{-1} can be used. Consider a two-dimensional aeroelastic system modeled by the following equations:

$$m\ddot{h} + s_\alpha\ddot{\alpha} + k_h h = -L(t), \qquad I\ddot{\alpha} + s_\alpha\ddot{h} + k_\alpha(t)\alpha = M(t) \quad (2)$$

where m is the mass, I is the mass moment of inertia, α is the pitch angle, s_α is the first moment of inertia, h is the plunge displacement, and k_h and k_α are the stiffness in plunge and pitch, respectively. In state-space form, this system can be rewritten as

$$\{\dot{x}\} + [C(t)]\{x\} = \{f\} \quad (3)$$

Here

$$C(t) = \begin{pmatrix} 0 & -1 & 0 & 0 \\ -\dfrac{k_h s_\alpha}{mI - s_\alpha^2} & 0 & \dfrac{k_\alpha m}{mI - s_\alpha^2} & 0 \\ 0 & 0 & 0 & -1 \\ \dfrac{k_h I}{mI - s_\alpha^2} & 0 & -\dfrac{k_\alpha s_\alpha}{mI - s_\alpha^2} & 0 \end{pmatrix}$$

$$f(t) = \begin{pmatrix} \dfrac{LI - Ms_\alpha}{mI - s_\alpha^2} \\ \dfrac{Mm - Ls_\alpha}{mI - s_\alpha^2} \end{pmatrix}$$

Using the method of matrix exponential time marching, the solution to this system for a time interval is written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\exp\left\{\int_{t_0}^{t}[C(t)]\,\mathrm{d}t\right\}\{x\}\right) = \exp\left\{\int_{t_0}^{t}[C(t)]\right\}f \quad (4)$$

if the noncommutative matrix multiplication between

$$\exp\left\{\int_{t_0}^{t}[C(t)]\,\mathrm{d}t\right\}$$

and $[C(t)]$ does not contribute a significant error. In such a case, this system can be solved as

$$x(t) = \exp\left\{-\int_{t_0}^{t}[C(t)]\,\mathrm{d}t\right\} \times \left[\{x_0\} + \int_{t_0}^{t}\left(\exp\left\{\int_{t_0}^{t}[C(\tau)]\,\mathrm{d}\tau\right\}\{f\}\right)\mathrm{d}t\right] \quad (5)$$

Here the time interval (t_0, t), chosen for each solution in the time-marching scheme, is fixed with respect to minimizing the errors because of the noncommutative matrix multiplication, that is,

$$\exp\left\{\int_{t_0}^{t}[C(t)]\,\mathrm{d}t\right\}[C(t)] - [C(t)]\exp\left\{\int_{t_0}^{t}[C(t)]\,\mathrm{d}t\right\} = \epsilon(t) \quad (6)$$

where $\epsilon(t)$ is the error tolerance level. Equation (5) can be written as

$$x_{n+1} = \mu_{n+1}(x_n + \lambda_{n+1}) \quad (7)$$

where

$$\mu_{n+1} = \exp\left\{-\int_{t_n}^{t_{n+1}}[C(t)]\,\mathrm{d}t\right\}$$

$$\lambda_{n+1} = \int_{t_n}^{t_{n+1}}\left(\exp\left\{\int_{t_n}^{t_{n+1}}[C(\tau)]\,\mathrm{d}\tau\{f\}\right\}\right)\mathrm{d}t$$

This means the response at time t_{n+1} is linked to the response at time t_n by the state transition matrix at time t_{n+1}, namely, μ_{n+1} and the forcing integral vector λ_{n+1}.

To simulate the aeroelastic behavior of a time-variant two-dimensional system, three neural networks are separately trained and then interlinked based on the method of matrix exponential time marching. We define the following input-output relationship for the three neural networks:

1) ANN(1) Input, $(t_i \;\; t_{i+1} \;\; k_\alpha(t_{i+1}))$; Output, $[\mu]$.

2) ANN(2) Input, $(t_i \;\; t_{i+1} \;\; u_\infty \;\; h \;\; \dot{h} \;\; \alpha \;\; \dot{\alpha})$; Output, $(c_{l_i} \;\; c_{l_{i+1}} \;\; c_{m_i} \;\; c_{m_{i+1}})$.

3) ANN(3)Input, $(t_i \;\; t_{i+1} \;\; k_\alpha(t_{i+1}) \;\; c_{l_i} \;\; c_{l_{i+1}} \;\; c_{m_i} \;\; c_{m_{i+1}})$; Output, $\{\lambda\}$.

Aeroelastic Computations

The time step used for computing the unsteady aerodynamic pressure over the airfoil using the doublet-panel method is determined by the appropriate wake shedding frequency. The wake shedding frequency is matched by the comparison between the obtained panel method solution for unsteady lift and the theoretical Wagner solution. As seen in Fig. 1, an appropriate match between the two different methods was obtained. This time step of the value of 0.004 s is fixed for unsteady aerodynamic computations. The size of the time step used in matrix exponential time marching can be determined by Eq. (8). Using the Leibnitz rule for differentiation under an integral sign, one can determine that

$$\left.\frac{\partial\epsilon(t,\Delta)}{\partial\Delta}\right|_{t_0} = \epsilon(t_0)C(t_0) \quad (8)$$

Here Δ is the time-step size $t_1 - t_0$ and C is the matrix of the coefficients of the differential equations constituting Eq. (4). For the time steps used in matrix exponential time marching, that is, in the range of 0.05–0.1, the norm of the error matrix, that is, $\epsilon(t)$, is of the order of 10^{-4}. Hence the change in this error with time-step size Δ is given in Eq. (8) to be determined by the product of the error matrix with the coefficient matrix. The terms of $C(t)$ involve ratios between products of the stiffness and inertias and hence are of the order of 10^1 to 10^2 for the analysis being conducted here. The eigenvalues of $C(t)$ are proportional to the square root of these terms. Thus the sensitivity term $[\partial\epsilon(t,\Delta)/\partial\Delta]|_{t_0}$ as expressed in Eq. (8) does not vary significantly with changes in time step Δ because the norm of ϵ is of the order of 10^{-4}.

Thus the neural network ANN1, trained using the method matrix exponential time marching, does not show high sensitivity to time-step size changes. However the neural network ANN2 representing the unsteady aerodynamic load requires a fixed time-step size. The time step used in training ANN2 is limited to evaluating the integral in Eq. (5), and hence it is a local time-step size, that is, the number of points chosen in the numerical integration is determined by the time step required for modeling the aerodynamic loads. The global time step, that is, the time step used when interlinking the three trained neural networks, is still determined by using the method of matrix exponential time marching.

Neural-Network Modeling for Flutter Suppression

The Levenberg–Marquardt routine is used to train the three neural networks titled ANN1, ANN2, and ANN3 to represent the dynamic aeroelastic system. The networks are designed with respect to both a fixed system stiffness and a varying torsional stiffness. The variation in the torsional stiffness is according to Eq. (1). The fixed stiffness neural-network system will represent the corresponding aeroelastic system exactly within the tolerance bounds because the training routine uses the same system parameters. However, for a varying stiffness system, it is not possible to train the neural-network system for all possible stiffness variations. Thus testing the neural networks for different stiffness variations is necessary. Accordingly, during the training process Eq. (1) uses certain constants c_1 and c_2, and during testing the constants will be changed. This tests the fidelity of the system to different system parameters.

The three neural networks are trained separately and then linked together, that is, ANN1 is trained using only the structural dynamics code, which can again be formulated using matrix exponential time marching, ANN2 is trained using the doublet-panel code, and ANN3

is trained based on integrating the forces that are obtained using the doublet-panel code. Having linked these three trained networks, we test the ANN at a system stiffness different from those used during training. Here we consider $c_1 = 1.33$, $c_2 = 1$, $k_0 = 5$, $k_{01} = 12$ for the testing of the ANNs. Note that the parameters c_1 and c_2 are different from those used for training the ANN. Figure 2 displays the result of the testing. The aeroelastic system as represented by this array of neural networks does show some differences from the numerical aeroelastic system, but these errors reduce with time. Moreover, the overall behavior of the neural-network system follows the numerical solver quite accurately.

Consider Fig. 3, which shows the three designed neural networks linked together to describe the aeroelastic system. A feedback loop is also shown, which evaluates the stability of the system. The stability is based on the response of the system during one period of oscillation. In the state-space plot, if the $|x^2 + \dot{x}^2|_{t+T} > |x^2 + \dot{x}^2|_t$, the system is unstable. Here, the period of oscillation is T, and the stability is calculated at a time step $t + T$. Because this is a linear system, this approach is valid. In general, the procedure to detect flutter is to calculate the work done over a period of oscillation by the external nonconservative forces. Hence the system is at the flutter boundary if

$$\triangle W_{\text{nc}} = \int_t^{t+T} \{f(t)\}^T \{\dot{x}\}\, \mathrm{d}t = 0 \tag{9}$$

where $f(t)$ is the nonconservative aerodynamic loads and $\{\dot{x}\}$ is the vector containing the system velocities. In Fig. 4, the initial dynamic aeroelastic response is calculated using the constant stiffness structural neural networks, ANN1 and ANN3. Once the system is tested to be unstable ($\triangle W_{\text{nc}} > 0$), the variable stiffness adaptive neural networks are switched on for the structural dynamics so as to bring the system again within the stability boundary. To achieve this, the neural-network array that was developed and tested in Fig. 2 is used. The reduced frequency of the aeroelastic system is set to 1.1, which is just over the flutter reduced frequency for this system. This can be seen in Fig. 4a. The neural-network controller detects the system instability after the first period and then switches over to the variable stiffness mode. The stiffness is now increased until stability is achieved. This is shown in Fig. 4b, which depicts a stable oscillatory pattern for the time history of the airfoil's pitching angle. For Fig. 4b, the stiffness variation that was used was $k_\alpha(t) = 5 + 12(1 - 1.33e^{-t})$. This was found to be sufficient to restore stability within one period of oscillation of the aeroelastic system.

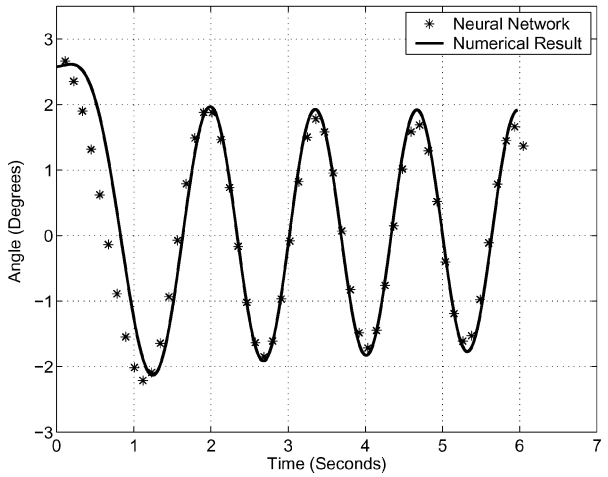


Fig. 2 Neural-network testing with variable torsional stiffness.

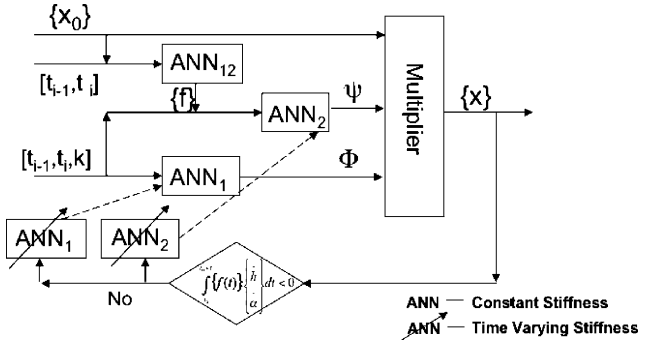
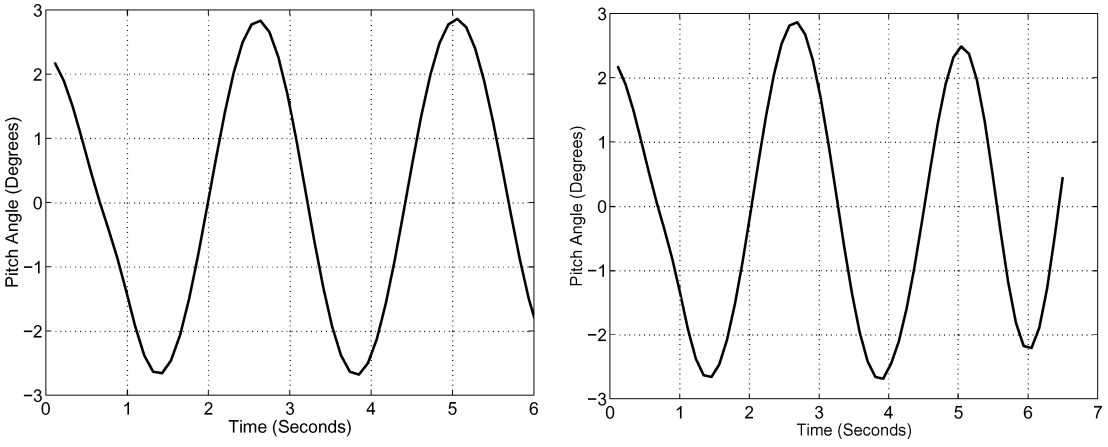


Fig. 3 Array of neural networks depicting both the constant stiffness aeroelastic system and the variable stiffness aeroelastic system.

The training of the neural networks using the MATLAB™ software on an SGI origin machine takes around 15 min of computational time using one processor. The computational time required for the neural-network simulation is extremely small. For example the 5-s simulation shown in Fig. 4a was obtained in a few seconds on an SGI Origin machine using one processor while running Matlab™. Because the neural-network array allows such a rapid simulation, the controller can directly calculate the response to detect instability and prevent it using the ANN based on the variable stiffness approach.

Conclusions

A neural-network-based aeroelastic response solver was developed for a two-dimensional dynamic aeroelastic system with a time-varying torsional stiffness. The trained neural networks described the free vibration response, the unsteady aerodynamic loads, and the forced response of the system. Hence when these three neural networks were integrated, the true dynamic aeroelastic response is obtained. The neural networks were trained from a doublet-panel method to describe the unsteady aerodynamic loads over an airfoil



a) b)

Fig. 4 Control of flutter at the flutter boundary using variable torsional stiffness: a) unstable pitching oscillations and b) flutter control by varying the torsional stiffness as $k_\alpha(t) = 5 + 12(1 - 1.33e^{-t})$.

and from a matrix exponential time-marching structural dynamics solver to evaluate the dynamic response of the system to these aerodynamic loads. The trained neural networks could function at a varying torsional stiffness within a specified range of stiffness variation. This array of neural networks was utilized to detect flutter in the aeroelastic system and suppress flutter by using the concept of a time-varying torsional stiffness.

References

[1]Gern, F. H., Inman, D. J., and Kapania, R. K., "Structural and Aeroelastic Modeling of General Planform Wings with Morphing Airfoils," *AIAA Journal*, Vol. 40, No. 4, 2002, pp. 628–637; also AIAA Paper 2001-1369, April 2001.

[2]Chen, P. C., Sarhaddi, D., Jha, R., Liu, D. D., Griffin, K., and Yurkowich, R., "Variable Stiffness Spar Approach for Aircraft Maneuver Enhancement Using ASTROS," *Journal of Aircraft*, Vol. 37, No. 5, 2000, pp. 865–871.

[3]Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., *Aeroelasticity*, Dover, New York, 1996, pp. 281–286.

[4]Natarajan, A., Kapania, R. K., and Inman, D. J., "Near-Exact Analytical Solutions of Linear Time-Variant Systems," *AIAA Journal*, Vol. 40, No. 11, 2002, pp. 2362–2366; also AIAA Paper 2001-1295, Nov. 2001.

[5]Raisinghani, S. C., and Ghosh, A. K., "Parameter Estimation of an Aeroelastic Aircraft Using Neural Networks," *Sadhana—Academy Proceedings in Engineering Sciences*, Vol. 25, Pt. 2, edited by N. Mukunda, Indian Academy of Sciences, Bangalore, India, 2000, pp. 181–191.

[6]Scott, R. C., and Pado, L. E., "Active Control of Wind-Tunnel Model Aeroelastic Response Using Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 6, 2000, pp. 1100–1108.

[7]Natarajan, A., Kapania, R. K., and Inman, D. J., "Aeroelastic Optimization of Adaptive Bumps for Yaw Control," *Journal of Aircraft*, Vol. 41, No. 1, 2004, pp. 175–185; also AIAA Paper 2002-0707, Jan. 2002.

[8]Ogata, K., *Discrete Time Control Systems*, Prentice–Hall, Upper Saddle River, NJ, 1987, Chap. 4.

A. Chattopadhyay
Associate Editor

Intergrid Transformation for Aircraft Aeroelastic Simulations

K. J. Badcock,* A. M. Rampurawala,† and B. E. Richards‡
University of Glasgow,
Glasgow, Scotland G12 8QQ, United Kingdom

Introduction

COMPUTATIONAL aeroelasticity requires the coupling of structural and fluid codes. Usually the structural grid and the fluid surface grids do not coincide, and hence a data exchange method is required that interfaces the loads and deformation information between the two grids. A review of some of these transformation methods is available in Hounjet and Meijer,[1] Guruswamy,[2] Smith et al.,[3] and Potsdam and Guruswamy.[4] The simulation of the aeroelastic response of wing-alone cases has become common.

Presented as Paper 2003-3512 at the 21st Applied Aerodynamics Conference, Orlando, FL, 23–26 June 2003; received 20 August 2003; revision received 20 February 2004; accepted for publication 17 April 2004.

*Reader, Computational Fluid Dynamics Laboratory, Department of Aerospace Engineering.

†Ph.D. Student, Computational Fluid Dynamics Laboratory, Department of Aerospace Engineering.

‡Mechan Professor, Computational Fluid Dynamics Laboratory, Department of Aerospace Engineering. Associate Fellow AIAA.

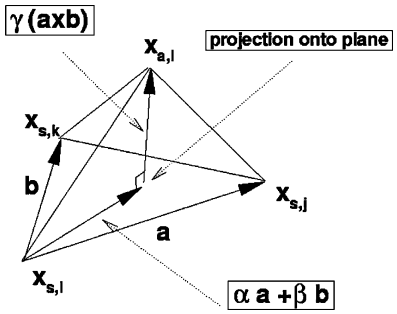


Fig. 1 Constant-volume-tetrahedron transformation.

Simulation in the transonic flow regime requires nonlinear aerodynamic codes, which in turn are sensitive to changes in external geometry. For a complete aircraft, multicomponents need to be transformed without introducing holes in the aerodynamic surface. A reliable transformation method for aircraft geometries is proposed in this Note and is demonstrated for the structural dynamics model of Huang and Zan,[5] a generic model for the F-16 aircraft.

Constant-Volume Tetrahedron

The constant-volume-tetrahedron (CVT) scheme is a transformation technique proposed in Goura[6] and Goura et al.[7] A surface element consisting of the three nearest structural grid points $\boldsymbol{x}_{s,i}(t)$, $\boldsymbol{x}_{s,j}(t)$, and $\boldsymbol{x}_{s,k}(t)$ to a given fluid grid point $\boldsymbol{x}_{a,l}(t)$ is identified. Once the structural grid points are identified and associated with the fluid grid point, the position of $\boldsymbol{x}_{a,l}$ is given by the expression

$$\boldsymbol{x}_{a,l} - \boldsymbol{x}_{s,i}(t) = \alpha\boldsymbol{a} + \beta\boldsymbol{b} + \gamma\boldsymbol{d} \tag{1}$$

where $\boldsymbol{a} = \boldsymbol{x}_{s,j} - \boldsymbol{x}_{s,i}$, $\boldsymbol{b} = \boldsymbol{x}_{s,k} - \boldsymbol{x}_{s,i}$, and $\boldsymbol{d} = \boldsymbol{a} \times \boldsymbol{b}$. Here the term $\alpha\boldsymbol{a} + \beta\boldsymbol{b}$ represents the location of the projection of $\boldsymbol{x}_{a,l}$ onto the structural triangle and $\gamma\boldsymbol{d}$ is the component out of the plane of this triangle, as shown in Fig. 1. In the preceding the values of α, β, and γ are calculated as

$$\alpha = \frac{|\boldsymbol{b}|^2(\boldsymbol{a}\cdot\boldsymbol{c}) - (\boldsymbol{a}\cdot\boldsymbol{b})(\boldsymbol{b}\cdot\boldsymbol{c})}{|\boldsymbol{a}|^2|\boldsymbol{b}|^2 - (\boldsymbol{a}\cdot\boldsymbol{b})(\boldsymbol{a}\cdot\boldsymbol{b})} \tag{2}$$

$$\beta = \frac{|\boldsymbol{a}|^2(\boldsymbol{b}\cdot\boldsymbol{c}) - (\boldsymbol{a}\cdot\boldsymbol{b})(\boldsymbol{a}\cdot\boldsymbol{c})}{|\boldsymbol{a}|^2|\boldsymbol{b}|^2 - (\boldsymbol{a}\cdot\boldsymbol{b})(\boldsymbol{a}\cdot\boldsymbol{b})} \tag{3}$$

$$\gamma = \frac{(\boldsymbol{c}\cdot\boldsymbol{d})}{|\boldsymbol{d}|^2} \tag{4}$$

As the structure deforms, the values of the structural grid points change. The location of the fluid grid point is recalculated using Eq. (1) with the following assumptions. First, the projection of the fluid point is forced to move linearly in the structural triangle by fixing α and β at their initial values. The value of γ, which scales the out-of-plane component, is calculated to ensure that the volume of the tetrahedron formed by the three structural and one fluid points remains constant. If the fluid and the structural points are planar, then the expression reduces to linear interpolation for the position of the fluid point.

This method has previously been tested for the aeroelastic response of isolated wings in Goura et al.[8] The extension to complete aircraft configurations is considered in the following section.

Transformation for Complete Aircraft

A version of the CVT that can do the transformation for a complete aircraft with the minimum of manual intervention and that preserves the surface mesh, particularly at junctions between components, is required. The insight for the method is provided by the paper of Melville,[9] which treats the aircraft components in a hierarchy.

The first stage of the method is to partition the fluid and structural points into levels associated with components. The primary component is the fuselage because all of the other parts of the aircraft