Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Sound and Vibration 275 (2004) 719–738
JOURNAL OF SOUND AND VIBRATION
www.elsevier.com/locate/jsvi

Maneuvering rotorcraft noise prediction ☆

G.A. Brès[a], K.S. Brentner[a,*], G. Perez[a], H.E. Jones[b]

[a] *Department of Aerospace Engineering, The Pennsylvania State University, University Park, PA 16802, USA*
[b] *U.S. Army Aeroflightdynamics Directorate, AMRDEC, U.S. Army Aviation and Missile Command, NASA Langley Research Center, Hampton, VA 23681-0001, USA*

Received 8 July 2002; accepted 2 July 2003

Abstract

This paper presents the unique aspects of the development of an entirely new maneuver noise prediction code called PSU-WOPWOP. The main focus of this work is development of a noise prediction methodology, which will enable the study of the aeroacoustic aspects a rotorcraft in maneuvering flight. It is assumed that the aeromechanical data (namely aircraft and blade motion, blade airloads) are provided as input data. This new noise prediction capability was developed for rotors in steady and transient maneuvering flight. Featuring an object-oriented design, the PSU-WOPWOP code allows great flexibility for complex rotor configuration and motion (including multiple rotors and full aircraft motion). The relative locations and number of hinges, flexures, and body motions can be arbitrarily specified to match any specific rotorcraft. An analysis of algorithm efficiency was performed for maneuver noise prediction along with a description of the tradeoffs made specifically for the maneuvering noise problem. Noise predictions for the mainrotor of a rotorcraft in steady descent, transient (arrested) descent, hover and a "pop-up" maneuver are demonstrated.

1. Introduction

Over the past decade, there has been a sustained interest in rotor noise prediction for reasons ranging from noise reduction required by stricter noise standards for civil aircraft to greater stealth in a military environment. A great deal of progress has been made in fundamental theoretical understanding and computational accuracy of both impulsive and non-impulsive noise

☆ Presented at the AHS Aerodynamics, Acoustics, and Test Evaluation Specialist Meeting, San Francisco, CA, January 23–25, 2002.

*Corresponding author. Tel.: +1-814-865-6433; fax: +1-425-963-2383.
E-mail address: ksbrentner@psu.edu (K.S. Brentner).

0022-460X/$ - see front matter
doi:10.1016/j.jsv.2003.07.005

sources, but evaluation of noise from maneuvering rotorcraft remains a largely untackled and extremely challenging problem. In both civil and military operations, real-world helicopters must maneuver with complex motions, including unsteady, non-periodic conditions, transient effects, pitch, roll and yaw motions: the noise generated is potentially very different from the current steady analysis results. Only recently has the prediction of the noise of a maneuvering rotorcraft been addressed, and no current method fully models even the simplest maneuvers.

The prediction of maneuvering rotorcraft noise brings with it new computational challenges: (1) rather than an isolated rotor, the complete rotorcraft must be considered to determine its flight path and rotor blade motion; (2) the blade loading and motion can be non-periodic during a maneuver and distinct for each rotor blade; and finally (3) the time scale of even a short maneuver is much greater than a single-blade passage, hence the noise prediction must encompass a much longer period of physical time. Current rotor noise prediction codes (e.g., WOPWOP [1], RAPP [2], PARIS [3], etc.) typically model only the noise from steady aircraft motion, hence, there is a need for a new code using an algorithm adapted for maneuver noise prediction. Furthermore, the opportunity exists in writing a new code to design it for the unique attributes of the maneuver problem—including the capability to analyze a very wide array of rotorcraft design features; take advantage of new advanced algorithm innovations for greater computational efficiency; and utilize modern object oriented code design to maximize code flexibility. The first goal of the paper is to outline the algorithm analysis and the development of this new code called PSU-WOPWOP. An algorithm analysis was made (and is presented) to quantitatively evaluate the efficiency of alternative algorithms for the maneuver noise problem. A secondary goal of the paper is to demonstrate the code's capabilities through the prediction of noise for both steady and transient flight conditions. Comparison with the WOPWOP [1,4] noise prediction code is also presented as a first step in validation. Although the code utilizes the theory behind WOPWOP, it is an entirely new code.

2. Algorithm analysis

2.1. Integral formulation

The typical starting point for rotor noise prediction is one of the various forms of the solution to Ffowcs Williams–Hawkings (FW–H) equation [5]. For the PSU-WOPWOP implementation, the FW–H equation for a permeable surface was utilized in the form of the integral representation of the solution known as formulation 1A [6,7]:

$$p'(\vec{x},t) = p'_T(\vec{x},t) + p'_L(\vec{x},t), \tag{1}$$

where p' is the acoustic pressure, $\vec{x}$ the observer position, t the observer time, and the subscript T and L correspond to thickness and loading components, respectively, and where

$$4\pi p'_T(\vec{x},t) = \int_{f=0}\left[\frac{\rho_0(\dot{U}_n + U_{\dot{n}})}{r(1-M_r)^2}\right]_{ret} \mathrm{d}S + \int_{f=0}\left[\frac{\rho_0 U_n(r\dot{M}_r + c(M_r - M^2))}{r^2(1-M_r)^3}\right]_{ret} \mathrm{d}S \tag{2}$$

and

$$4\pi p'_L(\vec{x},t) = \frac{1}{c}\int_{f=0}\left[\frac{\dot{L}_r}{r(1-M_r)^2}\right]_{ret} \mathrm{d}S + \int_{f=0}\left[\frac{L_r - L_M}{r^2(1-M_r)^2}\right]_{ret} \mathrm{d}S + \frac{1}{c}\int_{f=0}\left[L_r\frac{(r\dot{M}_r + c(M_r - M^2))}{r^2(1-M_r)^3}\right]_{ret} \mathrm{d}S. \quad (3)$$

The dot over a variable implies source-time differentiation of that variable, and a subscript r or n indicates a dot product of the vector with the unit vector in the radiation direction, $\hat{r}$, or outward surface normal direction, $\hat{n}$, respectively. The moving surface considered in the integration is defined by the function $f(\vec{x},t) = 0$, $\vec{M}$ is the local surface velocity vector divided by the free-stream sound speed. The subscript *ret* denotes that the integrand is evaluated at the retarded time. The vector components U_i and L_i are defined as

$$U_i = \left[1 - \frac{\rho}{\rho_0}\right]v_i + \frac{\rho}{\rho_0}u_i, \quad (4)$$

$$L_i = P_{ij}\hat{n}_j + \rho u_i(u_n - v_n). \quad (5)$$

2.2. Code design

Several decisions were made to design and adapt the code to handle maneuver cases efficiently. First, the permeable surface formulation of the FW–H equation (Eqs. (1)–(5)) was chosen so that transonic effects can be included if the input data are available. In this case the integration surface does not coincide with the blade surface but surrounds the blade, thus including any non-linear effects inside the permeable integration surface. Utilization of the formulation in this way enables close coupling with CFD for high-speed impulsive noise computation. Application of the formulation on the actual blade surface results in the more traditional form of the FW–H equation. Second, the rotor blade surface is assumed to be rigid in the acoustics code to simplify the blade motion computation. Nevertheless, elastic blade motion is properly modelled in the aerodynamics analysis; hence the primary effect of blade elasticity is included in the noise prediction through the blade loading and motion. The primary error associated with the rigid-blade acoustic model is in the computation of the radiation vector (i.e., radiation distance r and direction $\hat{r}$). This error is often negligible in typical noise predictions. Finally, an object-oriented approach was chosen to reduce programming errors, increase modularity, and provide flexibility for implementing complex rotor configurations. One new feature of PSU-WOPWOP is the use of task specific data structures objects described later in the paper, such as "rotor," "blade," or "patches." This approach takes advantage of the modern programming practice and enables the users to efficiently handle any number of rotors, in any arbitrary configuration, for any motion.

To represent the rigid-body blade motion (rotation, flapping, lead-lag, pitch, etc.) and the complete aircraft motion (non-periodic, time-dependent aircraft pitch, roll, yaw, etc.), a series of co-ordinate transformations from an observer-fixed frame of reference through a set of intermediate reference frames are needed. Namely, all the vector components in the integrand calculation must be expressed in the same frame of reference (i.e., the observer frame). This problem was considered using the mathematics of a multi-body dynamics problem with many

frames of reference, each simple motion leading to a new frame. For the noise computation, the position, velocity and acceleration of each point on the blade, at each source time τ, are required. To compute the position of a point with co-ordinates belonging to a frame in terms of the co-ordinates of a previous frame, a simple matrix algebra relation is used:

$$\vec{y}_{i-1} = [\mathbf{T}_{i/i-1}(\tau)]\vec{y}_i, \tag{6}$$

where $[\mathbf{T}_{i/i-1}]$ is the general transformation matrix relating the frame i to the frame $i-1$. The transformation matrix $[T_{i/i-1}]$ can be thought of as a 4×4 matrix that accounts for both rotation and translation of the reference frame, e.g.,

$$[T_{i/i-1}] = \begin{bmatrix} R_{11} & R_{12} & R_{13} & x_0 \\ R_{21} & R_{22} & R_{23} & y_0 \\ R_{31} & R_{32} & R_{33} & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \vec{y} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix},$$

where R_{ij} are the elements of the rotation matrix and x_0, y_0, z_0 are the translations from the $i-1$ frame to the i frame in the respective co-ordinate directions.[1] Therefore, if $\vec{\eta}$ is a position vector of a point in the blade-fixed frame F_N, then the position vector of the point in the observer-fixed frame F_0 at time τ is simply

$$\vec{y}(\vec{\eta}, \tau) = [\mathbf{T}_{N/0}(\tau)]\vec{\eta}, \tag{7}$$

where $[\mathbf{T}_{N/0}(\tau)] = [\mathbf{T}_{1/0}(\tau)]\cdots[\mathbf{T}_{i/i-1}(\tau)]\cdots[\mathbf{T}_{N/N-1}(\tau)]$ and the ellipsis denote that not all matrix multiplications are shown.

In addition to the position of the source, the acoustic formulation requires velocity and acceleration of the source. Eq. (7) can be differentiated once to determine the velocity, i.e.,

$$\begin{aligned} \vec{v}(\vec{\eta}, \tau) = {} & [\dot{T}_{1/0}]\cdots[T_{i/i-1}]\cdots[T_{N/N-1}]\vec{\eta} + \cdots + [T_{1/0}]\cdots[\dot{T}_{i/i-1}]\cdots[T_{N/N-1}]\vec{\eta} \\ & + \cdots + [T_{1/0}]\cdots[T_{i/i-1}]\cdots[\dot{T}_{N/N-1}]\vec{\eta} + [T_{N/0}]\frac{\mathrm{d}\vec{\eta}}{\mathrm{d}\tau} \end{aligned} \tag{8}$$

and a second time to determine the acceleration, i.e.,

$$\begin{aligned} \vec{a}(\vec{\eta}, \tau) = {} & [\ddot{T}_{1/0}][T_{N/1}]\vec{\eta} + \cdots + [\dot{T}_{1/0}][T_{1/i-1}][\dot{T}_{i/i-1}][T_{N/i}]\vec{\eta} + \cdots + [\dot{T}_{1/0}][T_{N/1}]\frac{\mathrm{d}\vec{\eta}}{\mathrm{d}\tau} \\ & + \cdots + [\dot{T}_{1/0}][T_{i-1/1}][\dot{T}_{i/i-1}][T_{N/i}]\vec{\eta} + \cdots + [T_{i-1/0}][\ddot{T}_{i/i-1}][T_{N/i}]\vec{\eta} \\ & + \cdots + [T_{i-1/0}][\dot{T}_{i/i-1}][T_{N/i}]\frac{\mathrm{d}\vec{\eta}}{\mathrm{d}\tau} + \cdots + [\dot{T}_{1/0}][T_{1/N-1}][\dot{T}_{N/N-1}]\vec{\eta} \\ & + \cdots + [T_{N-1/0}][\ddot{T}_{N/N-1}]\vec{\eta} + [T_{N-1/0}][\dot{T}_{N/N-1}]\frac{\mathrm{d}\vec{\eta}}{\mathrm{d}\tau} \\ & + [T_{N/0}]\frac{\mathrm{d}^2\vec{\eta}}{\mathrm{d}\tau^2}. \end{aligned} \tag{9}$$

[1] The actual implementation is more efficient because it does not perform the multiplications by zero or 1 indicated in the matrix multiplication $\vec{y}_{i-1} = [T_{i/i-1}]\vec{y}_i$. The use of 4×4 transformation matrices is common in computer graphics applications, in which case the operations may be implemented in hardware for extremely fast execution.

In these equations, the dot over the transformation matrix indicates a time differentiation. It is apparent in Eqs. (8) and (9) that the number of terms grows significantly during the process of differentiation. In fact, the number of operations required to determine the position, velocity, and acceleration are proportional to N, N^2, and N^3, respectively. Although this approach was used in WOPWOP [1], it is not efficient if there are a large number of reference frames.[2]

Instead of differentiating the co-ordinate transformation matrices to obtain velocity and acceleration, a method more suitable for the maneuver problem was selected. The formulation is used in the robotics industry to deal with multi-body dynamics problems and is significantly more computationally efficient. This method is based on "torsor" algebra (from the French word "torseur") instead of matrix algebra [8]. To compute the velocity of a point P on the blade surface, the Koenig (or Varignon) relation [9] is used. This formula relates the velocity of P to the velocity of the origin of the blade frame, O_N:

$$\vec{V}_{P\in N/0} = \vec{V}_{O_N\in N/0} + \vec{\Omega}_{N/0} \times \overrightarrow{O_N P}. \tag{10}$$

Here $\vec{V}_{P\in N/0}$ is the velocity of the point P considering the complete motion of the frame F_N (the blade frame) with respect to the frame F_0 (the observer frame); $\vec{V}_{O_N\in N/0}$ is the velocity of the point O_N considering the complete motion of the frame F_N with respect to the observer frame F_0; and $\vec{\Omega}_{N/0}$ is the rotation speed the blade frame F_N relative to the observer frame F_0. The relations

$$\vec{\Omega}_{N/0} = \vec{\Omega}_{1/0} + \cdots + \vec{\Omega}_{i/i-1} + \cdots + \vec{\Omega}_{N/N-1}, \tag{11}$$

$$\vec{V}_{O_N\in N/0} = \vec{V}_{O_N\in 1/0} + \cdots + \vec{V}_{O_N\in i/i-1} + \cdots + \vec{V}_{O_N\in N/N-1} \tag{12}$$

are used to calculate the velocity of the origin of the Nth frame ($\vec{V}_{O_N\in N/0}$) and the rotation speed $\vec{\Omega}_{N/0}$. The additions in Eqs. (11) and (12) presume that the components of all the vectors are specified in the same reference frame, which usually is the observer frame. The aircraft and blade dynamics are provided, hence the angular and linear velocity of the i frame relative to the $i-1$ frame ($\vec{\Omega}_{i/i-1}$ and $\vec{V}_{O_i\in i/i-1}$, respectively) are known. The velocity $\vec{V}_{O_N\in i/i-1}$ is determined from another application of the Koenig relation,

$$\vec{V}_{O_N\in i/i-1} = \vec{V}_{O_i\in i/i-1} + \vec{\Omega}_{i/i-1} \times \overrightarrow{O_i O_N} \tag{13}$$

and $\overrightarrow{O_i O_N}$ is the vector from the origin of the i frame to the origin of the N frame. With this, all the information necessary to compute the velocity of any point of the blade with Eq. (10) is available.

Similar relations are used to compute the acceleration. The acceleration of the point P is related to the acceleration of the origin of the blade frame O_N using the point change formula:

$$\vec{A}_{P\in N/0} = \vec{A}_{O_N\in N/0} + \frac{\mathrm{d}\vec{\Omega}_{N/0}}{\mathrm{d}\tau} \times \overrightarrow{O_N P} + \vec{\Omega}_{N/0} \times \vec{\Omega}_{N/0} \times \overrightarrow{O_N P}. \tag{14}$$

This formula enables the calculation of the acceleration of any point (here P) belonging to one frame of reference if the acceleration of one point in this frame is already known (here O_N). The subscript $P\in N/0$ means that the acceleration of a point P is calculated considering the complete motion of the N frame in the 0 frame. A recursion procedure can be used to calculate $\vec{A}_{O_N\in N/0}$.

[2] WOPWOP included between one and four co-ordinate transformations, whereas in a maneuver computation a large number of transformations (i.e., $N = 13$ or greater) is expected.

Suppose that all information for the $i-1$ frame is known; then the composition of acceleration relation enables us to combine the acceleration of each frame:

$$\vec{A}_{O_i \in i/0} = \vec{A}_{O_i \in i/i-1} + \vec{A}_{O_i \in i-1/0} + 2\vec{\Omega}_{i-1/0} \times \vec{V}_{O_i \in i/i-1}. \tag{15}$$

Here again the point change relation is needed to relate $\vec{A}_{O_i \in i-1/0}$ (unknown) to $\vec{A}_{O_{i-1} \in i-1/0}$ which has been calculated at the previous iteration:

$$\vec{A}_{O_i \in i-1/0} = \vec{A}_{O_{i-1} \in i-1/0} + \frac{\mathrm{d}\vec{\Omega}_{N/0}}{\mathrm{d}\tau} \times \overrightarrow{O_{i-1}O_i} + \vec{\Omega}_{i-1/0} \times \vec{\Omega}_{i-1/0} \times \overrightarrow{O_{i-1}O_i}, \tag{16}$$

where $\overrightarrow{O_{i-1}O_i}$ is the vector from the origin of the $i-1$ frame to the origin of the i frame. With Eqs. (15) and (16), it is possible to deduce $\vec{A}_{O_i \in i/0}$ from $\vec{A}_{O_i \in i/i-1}$, the acceleration of the i frame considering its relative motion with respect to the $i-1$ frame. The calculation of $\vec{A}_{O_N \in N/0}$ is performed using this recursion procedure; then an application of Eq. (14) is used to deduce the acceleration of any point on the blade. In this approach, N applications of the recursion (Eqs. (15) and (16)) are required. More details about the representation of velocity and acceleration can be found in Ref. [10].

This approach to the kinematics enabled us to significantly reduce the number of operations involved in the motion description: the operation counts for velocity and acceleration computations are proportional to number of transformations N, instead of N^2 and N^3, respectively, for matrix algebra. For example, matrix algebra requires $9N^3 + 15N$ operations to compute the acceleration of a point. In contrast, the new approach uses only $150N + 33$ operations to calculate the acceleration of a point on the integration surface. This method proves to be more efficient for maneuver cases, where N becomes large. For example, if $N = 13$ (as in the transient maneuver cases which follow) the operations count is reduced by a factor 10. Similar savings of computational effort are realized in the velocity. Complete details of the operation count comparison are given in Ref. [10].

2.3. Algorithm efficiency analysis

Once the acoustic formulation and motion description were determined, an analysis of the integration algorithm was performed. Although formulation 1A is a retarded-time formulation, there are several approaches to finding the acoustic pressure time history with this formulation. The most common method of solution, normally referred to as the "retarded-time" algorithm, is used in WOPWOP [1]. The solution procedure starts with the choice of the observer time t at which the solution is desired. For each point on the integration surface, the blade is iteratively repositioned to determine where that point was when the sound was emitted, hence satisfying the retarded-time equation, $\tau = t - |\vec{x} - \vec{y}(\tau)|/c$. The retarded-time τ is the time at which the sound reaching the observer at time t was emitted. Notice that in this equation the source position $\vec{y}(\tau)$ is a function of the retarded time.

An alternative to the retarded-time algorithm is to use the same formulation, but to fix the source time τ and determine when the sound from each point on the blade surface will reach the observer. Since the arrival time t will be different for each point, the time history of each point on the surface must be interpolated so that the contributions can be summed at the same observer time t. This forward calculation of time is referred to as a "source-time-dominant" algorithm

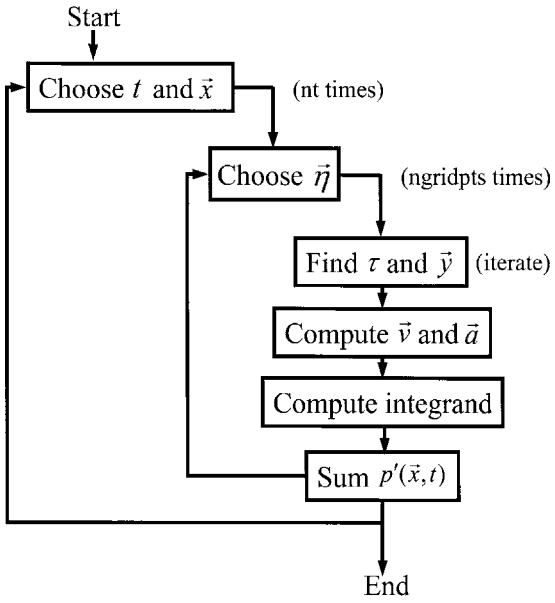

Fig. 1. Schematic of retarded-time algorithm.

[11].[3] In both the retarded-time and source-time-dominant approaches, once the appropriate time is found for a point on the blade surface, the integrand at that point may be determined, using Eqs. (2) and (3).

In both algorithms, the co-ordinate transformation matrices are needed to find the source position at the emission time. All transformation matrices $[T]$ are functions of the source time τ and independent of the observer time t. The computation costs, however, are significantly different for the two algorithms. In a retarded-time approach, $[T]$ must be computed for each point on the grid, for each observer time t—including an iteration to find the particular source τ that satisfies the retarded-time equation ($\tau = t - |\vec{x} - \vec{y}(\tau)|/c$). Fig. 1 shows a schematic of the retarded-time algorithm. On the other hand, the "source-time-dominant" algorithm only requires one evaluation of the co-ordinate transformation, for each source time τ, since the transformation matrices $[T]$ are the same for each point on the grid (τ is fixed). A schematic of the source-time-dominant algorithm is shown in Fig. 2.

The co-ordinate transformations are involved not only in the computation of the blade position, but also in the calculation of the source point velocity and acceleration as well. The number of floating point operations is a function of the number of co-ordinate transformations used. The source-time-dominant approach appears to be more efficient because co-ordinate transformations only need to be computed once for each source time, and a significant number of frames are expected. However, this approach contains two potential drawbacks. First, in the

[3] A variation of this approach in which zeroth order extrapolation is used has been called "binning technique" [12,13]. In this case, the signal is put in the closest "bin" and no interpolation is used to distribute the contribution of the source between the two desired arrival times nearest the computed arrival time.

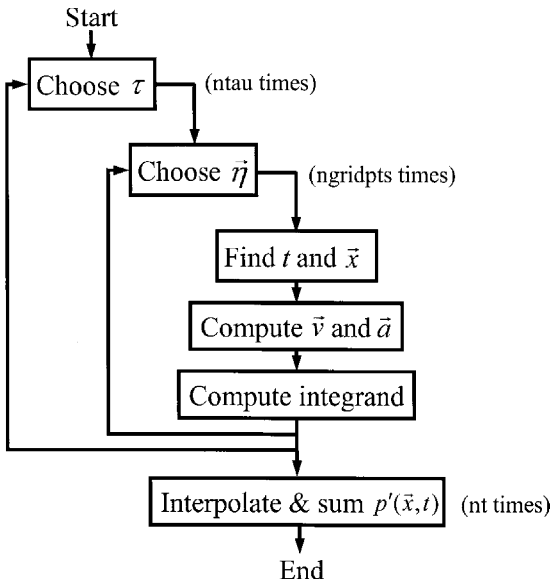

Fig. 2. Schematic of source-time-dominant algorithm.

source-time-dominant algorithm, the arrival time of the acoustic signal will be different for each point on the blade; hence, the time history for each point must be interpolated so that the contribution from each source point can be summed at the same observer time. This additional step is shown at the bottom of Fig. 2. Another potential issue is that more points may be required in the source-time history than in the acoustic-pressure time history ($n\tau \gg nt$). This would occur when high temporal resolution is required, such as in the case of blade–vortex interaction (BVI). Consequently, more integrand computations might be performed in the source-time-dominant approach, even though the number of co-ordinate transformation computations is significantly reduced.

To help clarify which algorithm is more efficient for computing the noise of maneuvering rotors, an analysis of operation counts for each of the two algorithms was performed and the important parameters identified (i.e., number of points on the integration surface—*ngridpts*, number of input data points in the source-time history—$n\tau$, number of output points in the observer-time history—nt, number of co-ordinate transformations, etc.). Experience with WOPWOP guided the analysis and, as expected, one of the most significant parameters was the number of co-ordinate transformations. The other parameters of particular importance for maneuver noise prediction were the number of points in the observer-time history and the ratio $nt/n\tau$. In Figs. 3 and 4, the height of the surface shows the expected operation count for both the retarded-time and source-time-dominant algorithms. The number of operations is plotted as a function of the number of co-ordinate transformations (right lower axis) and the number of points in the observer-time history nt (left lower axis). In Fig. 3, the ratio $nt/n\tau$ is equal to 3, which corresponds to a case where the observer-time resolution is greater than the source-time resolution. This is typical of the case for transient maneuver, with long time scale (large nt) and rather low-resolution time-dependent loading (data available only every 10° or so). From the

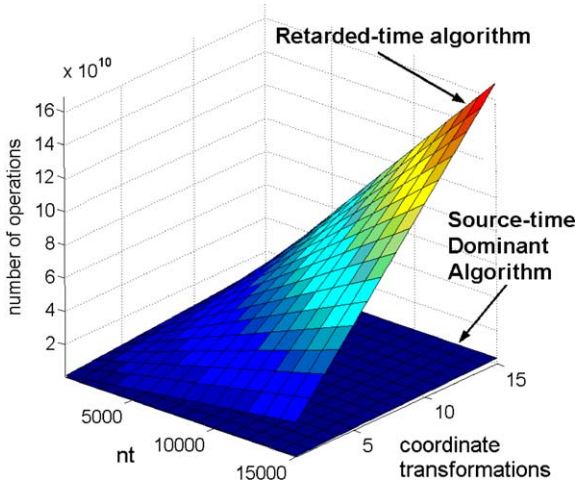

Fig. 3. Comparison of operation count for retarded-time and source-time-dominant algorithm for the ratio $nt/n\tau = 3$.

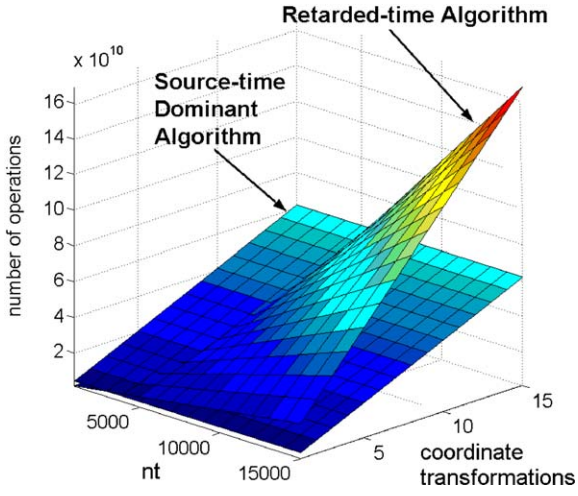

Fig. 4. Comparison of operation count for retarded-time and source-time-dominant algorithm for the ratio $nt/n\tau = 0.2$.

figure, it is clear that the source-time-dominant algorithm requires significantly fewer floating-point operations, especially for a large number of co-ordinate transformations. Fig. 4 shows the same comparison for a ratio $nt/n\tau$ equal to 0.2. The situation corresponds to a case where high-resolution loading data are available (e.g., data every half degree). This level of resolution is necessary for accurate computation of blade–vortex-interaction noise and other impulsive noise sources. Once again, the source-time-dominant approach is more efficient for maneuver even though the retarded-time algorithm requires fewer operations for a small number of points in the observer-time history. The number of co-ordinate transformations and the time scale are expected to be large for the prediction of maneuvering rotor noise (as many as 15 co-ordinate

transformations, up to 30-s time history). Therefore, based on this analysis, it was decided to use the source-time-dominant algorithm.

3. The maneuver code

3.1. Object-oriented design

The object-oriented design is one of the new features in PSU-WOPWOP. Several task-specific data structures together with associated functions define "objects" in the code implementation. Objects are organized in a hierarchy for convenience. The lowest level object is known as a "patch," which contains the two-dimensional surface definition of a piece of the blade surface, along with related quantities such as surface loading and normal vectors. Arrays of "patches" form objects known as "blades." Each "blade" contains additional information that is common among all the patches. Collections of "blades" are known as a "rotor" which also contains data relevant to all the blades. A configuration can contain several "rotor" objects. During implementation this hierarchy of objects helped reduce the number of coding errors and assisted testing operations. To implement the general motion formulation described previously, another type of data structure was introduced to efficiently characterize the co-ordinate transformation from one co-ordinate basis to another. An array of these co-ordinate transformations is used to construct each transformation needed for arbitrarily complex rigid-body motion. This approach gives users great flexibility to implement multiple rotors, multiple blades with any spacing or shape, and any arbitrary motion. More details of the PSU-WOPWOP code design can be found in Ref. [14].

3.2. Chordwise compact formulation

While the acoustic formulation described in Eqs. (1)–(5) assumes that the fluid pressure and velocity are provided all over the integration surface, often only the blade loading as a function of rotor radius is available from a comprehensive analysis. Brentner et al. [15] showed that a chordwise compact model for loading noise is reasonably accurate away from the rotor tip-path plane. Brentner and Jones [4] developed a chordwise compact loading noise formulation that can be written

$$4\pi p'_L(\vec{x},t) = \frac{1}{c}\int_{f=0}\left[\frac{\dot{L}_r}{r(1-M_r)^2}\right]_{ret} \mathrm{d}l + \int_{f=0}\left[\frac{L_r - L_M}{r^2(1-M_r)^2}\right]_{ret} \mathrm{d}l + \frac{1}{c}\int_{f=0}\left[L_r\frac{(r\dot{M}_r + c(M_r - M^2))}{r^2(1-M_r)^3}\right]_{ret} \mathrm{d}l. \quad (17)$$

In this formulation, $\vec{L}$ is the section loading vector, and l is the spanwise integration variable. Although the thickness noise still requires chordwise integration, this approximation makes the loading noise computation compatible with comprehensive analysis input data and significantly reduces the computational cost of the loading noise prediction. In the code, the compact loading noise is implemented as a "compact loading" patch object. Since thickness and compact loading noise are computed on separated patches, the source-time scales can be different for each patch.

This feature could be very efficient for BVI computation—source-time resolution must be extremely precise for loading to accurately capture BVI, while the thickness noise time discretization can be relatively coarse.

4. Code and methodology validation

4.1. Loading computation

The acoustic prediction requires the determination of the blade motion and loading. A comprehensive analysis code is ultimately required to provide the rotor trim conditions even if relatively sophisticated computational aerodynamic and structural dynamic codes are also used. If the comprehensive analysis code is used alone, then the questions of the adequacy of the dynamic stall, unsteady aerodynamics, and wake models are compounded by the complication of maneuvering flight. For this work, the CAMRAD 2 comprehensive analysis code [16] has been utilized.

In typical trim computations, a series of complicated non-linear algorithms are solved using highly damped versions of Newton's method (other methods are also available). The algorithms are "nested" within each other in a way to minimize run-time. Note also that various functions of the system interact at several different "levels" of the iteration. The control vector is systematically updated to achieve solution convergence. One of the more useful features of the CAMRAD 2 code is that it allows the user complete access to the numerical scheme. It is particularly easy, for example, to exercise complete control over damping convergence of the models.

An arrested descent maneuver was computed using the transient flight option available in the CAMRAD 2 model. Transient maneuvers are complex—neither blade motion nor loading are periodic. Thus, the typical assumption of periodicity is invalid. The forces and moments are integrated to obtain the response at each time step due to specified control input. A solution set from a previous trim is used to initiate the computation. For the present application, there are three things to note about this approach. First, since the solution is no longer periodic, it is necessary to treat each blade as a separate component. Second, the solution requires that the control settings be specified a priori. Third, the procedure does not routinely include a sub-iteration mechanism to stabilize the numerical integration—a shortcoming that leads to the accumulation of truncation, round-off and convergence errors that may eventually destroy a computation which is allowed to run too long. The transient used in the current example problem was chosen to be short in duration (~ 2 s) and hence avoided an accumulation of time-integration error.

Typically, the trim and maneuver algorithms rely heavily upon semi-empirical models of the various physical phenomena that must be addressed in the solution. A full discussion of all these models is beyond the scope of this paper but was presented by Johnson [16]. The key modelling assumptions employed in this work will now be reviewed. A notional four-bladed, articulated-rotor aircraft of approximately 62,300 N (14,000 lb) gross weight was used in this study. The main rotor blades have a radius of 7.32 m, with a 20° tip sweep initiated at approximately 95 span, approximately 8° linear twist and are modelled by two patches (upper and lower surface). The

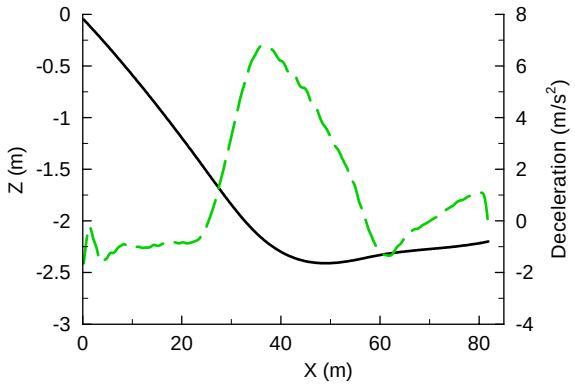

Fig. 5. Aircraft height (——) and vertical deceleration (— — —) during arrested descent maneuver.

computation was performed for a main rotor with four equally space blades. Since these computations are only intended to demonstrate the utility of PSU-WOPWOP code, details of the rotor and aircraft models were not chosen to correspond with any existing aircraft. The aircraft model includes main and tail rotors together with fuselage aerodynamics. Dynamic and structural responses for both rotors are computed. A rigid fuselage is assumed. The blade was modelled with 25 unequal length radial segments.

The transient maneuver calculation in CAMRAD 2 was performed with an integration time step of 0.002885 seconds ($\sim 5^\circ$ azimuth step) and the blade response time history was saved at 0.005770-s intervals ($\sim 10^\circ$ azimuth step). Blade loading computed at 5° azimuthal increments is insufficient to adequately characterize BVI, yet was deemed acceptable since the focus of this work is on the acoustic code development and not BVI noise prediction. The transient maneuver computation was fully time-dependent (not quasi-steady), but for simplicity both the main and tail rotor aerodynamics were computed using dynamic inflow rather than a more computationally intensive free wake. Lift and drag forces were resolved in a blade fixed system and were extracted for each blade as a function of time. Blade flap, lag, and pitch hinge displacements were also extracted as functions of time for each individual blade. The time history of the aircraft pitch, roll, and yaw orientations and the location of the aircraft center of gravity are computed and saved for the acoustic computation.

4.2. Transient test

To validate the new PSU-WOPWOP code, a comparison with WOPWOP predictions[4] is presented for a 3° arrested descent (also called a "pull up"). This maneuver is representative of a short-time transient maneuver, with both non-periodic blade motion and loading. The total time of the maneuver was 2 s; with a moderate helicopter forward speed of 40 m/s. Fig. 5 shows the

[4] The WOPWOP results are from Brentner and Jones [4]. Note: the version of WOPWOP used by Brentner and Jones was modified to perform maneuver computations specifically to determine the importance of transient maneuver on noise generation. This code has not been distributed.

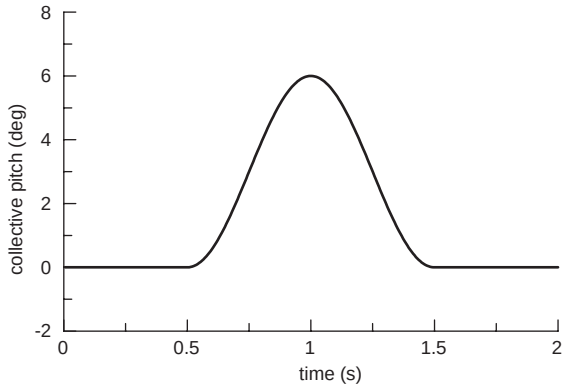

Fig. 6. Time history of collective pitch for the 3° arrested descent maneuver.

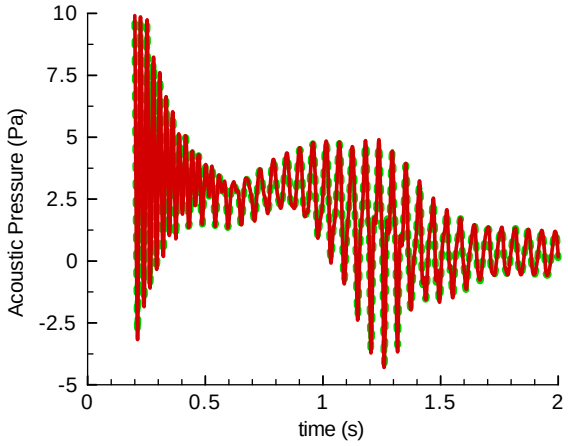

Fig. 7. Acoustic pressure comparison for a 3° arrested descent (with loading phase error): WOPWOP (– – – –); PSU-WOPWOP (——).

change of aircraft altitude and the rate of vertical deceleration. The descent of the helicopter was arrested using a collective pulse control input described in Fig. 6. The maximum vertical deceleration for this case was approximately 6.9 m/s^2 (1.7 g).

In this computation the observer was fixed and located 30.48 m (100 ft) below the helicopter center of gravity at source time $\tau = 0$. Therefore, the signal was received at the observer position shortly after the helicopter has flown over. Under these conditions, the loading noise happens to be dominant (thickness noise is completely negligible). The comparison between the PSU-WOPWOP and WOPWOP noise predictions is shown in Fig. 7. Even though both codes are based on the same formulation, the algorithms used are completely different, and an excellent agreement is obtained between the two codes. The computation time, however, is approximately 22 times faster for PSU-WOPWOP than the modified version of WOPWOP.

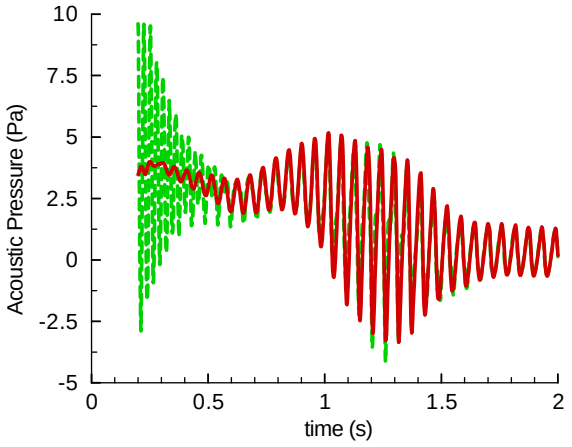

Fig. 8. Comparison of noise computed by PSU-WOPWOP with loading from uniform inflow and dynamic inflow models (with loading phase error): uniform inflow (– – – –): dynamic inflow (——).

The transient loading computation (arrested descent) of Brentner and Jones [4] was used as input for both of the noise predictions in Fig. 7. Two errors were found in the noise computations of Ref. [4] first, the loading computation was inadvertently based upon a uniform inflow model rather than the intended dynamic inflow model[5]; and second, the blade loading from CAMRAD 2 was indexed to the wrong blade resulting in a 90° phase error. For this paper, the 3° arrested descent was rerun with the dynamic inflow model. The new PSU-WOPWOP noise computation is compared with the previous result in Fig. 8. (The loading phase error is repeated to enable direct comparison with Fig. 7.) The most significant difference between these computations is the beginning of the time history—before the maneuver has begun. An investigation into the cause of the difference reveals one of the most challenging problems in computing the loading for a helicopter in maneuver: achieving stability and convergence of the time-dependent integration. The normal force time history at a radial station $r/R = 0.9325$ in Fig. 9 illustrates apparent numerical oscillations during the first one-half second of the computation. Notice in Fig. 9 that only the uniform inflow result contains high-frequency oscillations while the result from the dynamic inflow model is significantly smoother. This is not to imply that the uniform inflow model is fundamentally flawed, but rather that the numerical procedure used to transition from a steady case to a transient case is tricky—and in this example leads to oscillations. Although these oscillations appear to be entirely numerical in nature, they are of critical importance to the acoustic computation. The time derivative of the normal force at this radial station is shown in Fig. 10. Here the effect of the oscillations is very clear during the first one-half second (the "steady" part of the maneuver).

As a validation of the algorithm analysis used for the code PSU-WOPWOP code design, the execution time of WOPWOP is compared to PSU-WOPWOP in Table 1 for the arrested descent maneuver. All the computations were performed on a 1.7 GHz Pentium Xeon processor. As expected, there is a significant improvement in calculation times—a 55 times speed up—mainly

[5] A free-wake model was not used in order to minimize the complexity of the analysis for maneuver.

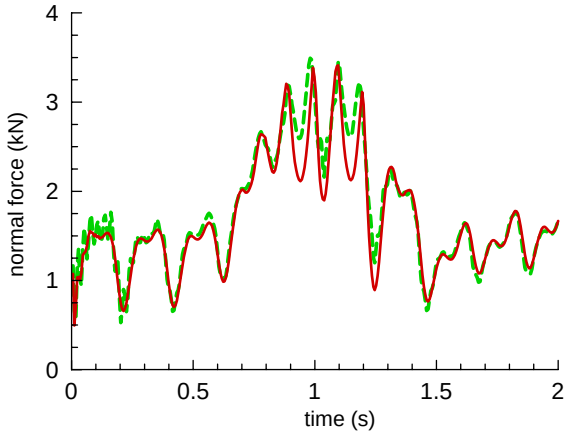

Fig. 9. Time history of normal force at $r/R = 0.9325$ for both uniform and dynamic inflow models: uniform inflow (– – – –); dynamic inflow (——).

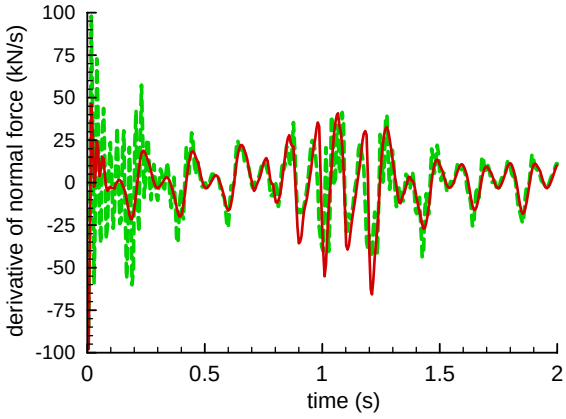

Fig. 10. Time history of normal force derivative at $r/R = 0.9325$ for both uniform and dynamic inflow models: uniform inflow (– – – –); dynamic inflow (——).

Table 1
Comparison of run-times on a 1.7 GHz Pentium Xeon processor

Arrested descent maneuver	Computation time (s) (thickness and loading noise)	Computation time (s) (loading noise only)
WOPWOP	315	
PSU-WOPWOP	5.73	1.21

due to the new code design and the run-time efficiency analysis performed. As mentioned earlier, WOPWOP uses derivatives of the transformation matrices for velocity and acceleration computation. Recall that with N co-ordinate transformations, the number of operations to compute velocity and acceleration are proportional to N^2 and N^3 instead of N, for each point on

the integration surface at each point in the observer-time history. The source-time-dominant approach in PSU-WOPWOP is particularly efficient for this case, which used 13 transformations to represent the full aircraft motion. Furthermore, in this computation thickness noise is negligible. Without the thickness noise computation, the code predicts the loading noise with an execution time of approximately 1.2 s—for a 1.8 s maneuver!

4.3. Acoustic impact of transient maneuver

Now, to demonstrate the impact of transient maneuver on the rotor noise, the noise generated during a 3° steady descent is compared with that of the transient (arrested descent) maneuver for the same helicopter/observer configuration. In Fig. 11, the acoustic pressure time history is shown for both operations—the blade loading is computed with the dynamic inflow model and the loading phase has been corrected. As expected, the results are identical during the first 0.6 s of the maneuver, which correspond to the steady part of the arrested descent. Then, the amplitude of the acoustic pressure increases significantly (by more than a factor of three) during the transient step corresponding to the increase of collective pitch to arrest the descent. The increased noise level during the maneuver is not only due to an increase in the rotor thrust (which increases roughly 70 percent during the maneuver), but is also due to increased local acceleration of the blade and unsteady loading which are accounted for through the time derivative terms in Eqs. (2) and (3). This additional noise is due to the transient nature of the maneuver and would not be fully reflected in a quasi-steady analysis.

A similar comparison of predicted noise was made between a hovering rotor and a "pop-up" maneuver. For both cases, the blade rotation speed is 293 r.p.m., and the observer is stationary at 30.48 m (100 ft) below the helicopter center of gravity and 30.48 m (100 ft) in front of it, at source time $\tau = 0$. Fig. 12 shows the change in aircraft altitude as a function of time while Fig. 13 shows the aircraft attitude during the "pop-up" maneuver. (Note: The tail rotor collective pitch was not adjusted during the maneuver, hence the aircraft yaws due to the addition of main rotor torque.) The total duration of the maneuver was 2 s and the maximum vertical acceleration is 7.8 m/s^2 (1.8 g). The same rotor configuration was used for these computations—four equally spaced

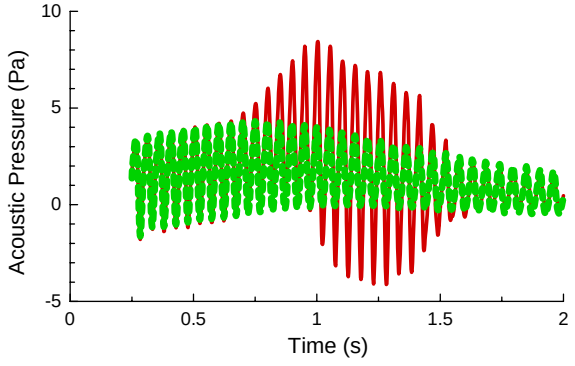

Fig. 11. Comparison of acoustic pressure time history computed with PSU-WOPWOP for a 3° steady descent (– – –) and an arrested descent maneuver (——).

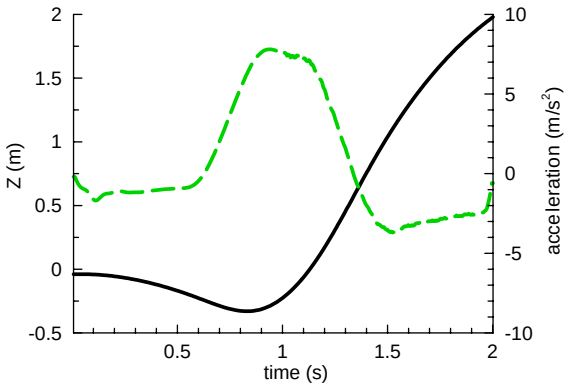

Fig. 12. Change in aircraft altitude (——) and vertical acceleration (— — —) during the pop-up maneuver.

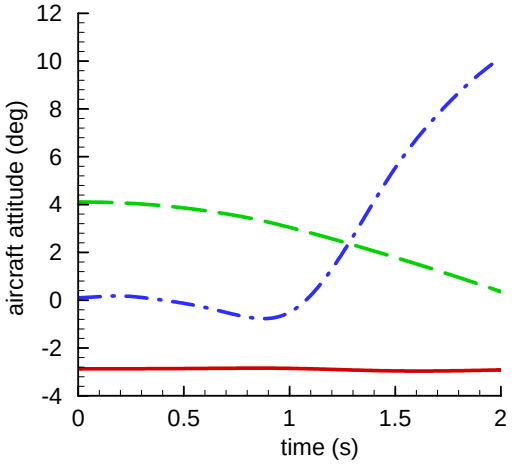

Fig. 13. Change in aircraft attitude during the pop-up maneuver: roll (——); pitch (— — —); yaw (·-·-·-·-·).

blades with the same blade geometry. Once again, the thickness noise is negligible. The comparison of the loading noise for each of these maneuvers is shown in Fig. 14. As expected, the acoustic pressure is periodic for the hover case, and matches exactly the first one-half second of the pop-up maneuver, where the altitude Z of the helicopter remains approximately constant, and equal to the hover altitude. The acoustic impact of the maneuver is then clearly identifiable, as the amplitude of the acoustic pressure in the "pop up" maneuver is nearly three times that of the hovering rotor even though the rotor thrust only increases by about 80 percent.

5. Concluding remarks

Although the prediction of the noise generated by a maneuvering rotor remains challenging, this paper begins the process of identifying the unique features of maneuver noise that require

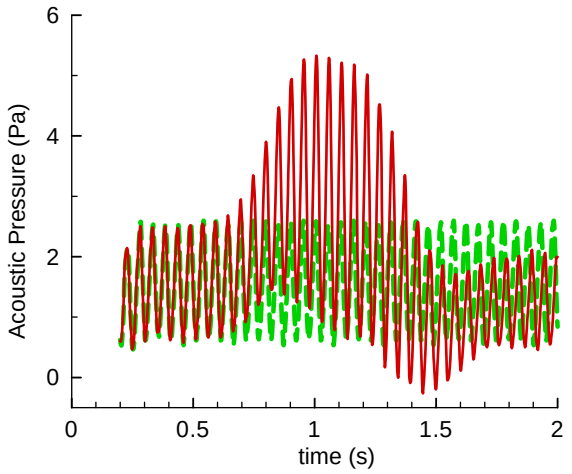

Fig. 14. Comparison of acoustic pressure time history computed with PSU-WOPWOP for a hover (– – – –) and "pop up" (——) maneuver.

appropriate acoustic modelling. The main focus of the paper was the design and development of a new code for maneuvering rotorcraft noise prediction called PSU-WOPWOP. Both steady and transient flight conditions can be modelled, with arbitrary aircraft and blade motions, for periodic or time-dependent data. The blade is assumed to be a rigid body, but otherwise, the relative locations and number of hinges, the blade shape and spacing can be specified to match any specific rotorcraft. The multiple rotor capability enables computation for any rotor configuration (i.e., main/tail rotor interaction, tilt rotors, etc.). The compact-chordwise loading formulation has been implemented to make the code more compatible with comprehensive analysis codes. The impact of maneuver on rotor noise radiation was demonstrated for a 3° arrested descent and a "pop up" maneuver—in each case the amplitude of the noise during the transient maneuver is significantly higher than for the related steady condition.

Acknowledgements

This research effort has been support by NASA Research Cooperative Agreement NCC-1-406 and the Penn State University NRTC Rotorcraft Center of Excellence (NASA Cooperative Agreement NAS2-36490, Task 4.1).

Appendix A. Nomenclature

c	sound speed in quiescent medium
$\mathrm{d}l$	element of the spanwise integration
$\mathrm{d}S$	element of the integration surface area
f	function defining the integration surface $f = 0$

L_i	components of vector defined in Eq. (5)
L_M	$L_i M_i$
L_r	$L_i \hat{r}_i$
$\dot{L}_r$	$\dot{L}_i \hat{r}_i$
$\vec{M}$	local Mach number vector of source
M_i	components of $\vec{M}$
M	$\lvert\vec{M}\rvert$
M_r	Mach number of source in radiation direction, $M_i \hat{r}_i$
$\hat{n}$	outward unit normal vector to surface
$\hat{n}_i$	components of $\hat{n}$
P_{ij}	compressive stress tensor with constant $\rho_0 \delta_{ij}$ subtracted
p	pressure
p'	acoustic pressure; $p - p_0$ outside source region
R	radius of the blade
r	distance between observer and source, $\lvert\vec{x} - \vec{y}\rvert$
$\hat{r}$	unit vector in radiation direction, $\hat{r} = (\vec{x} - \vec{y})/r$
$\hat{r}_i$	components of $\hat{r}$
t	observer time
$[T_{i/i-1}(\tau)]$	general transformation matrix relating frame i to frame $i-1$ at time τ
U_i	components of vector defined in Eq. (4)
U_n	$U_i \hat{n}_i$
$U_{\dot{n}}$	$U_i \dot{\hat{n}}_i$
$\dot{U}_n$	$\dot{U}_i \hat{n}_i$
u_i	components of local fluid velocity
u_n	$u_i \hat{n}_i$
$\vec{V}_{P \in i/i-1}$	velocity of point P of frame i into frame $i-1$
v_n	local normal velocity of source surface
X	first component of the helicopter position in the observer frame
$\vec{x}$	observer position vector
x_i	components of $\vec{x}$
$\vec{y}$	source position vector
y_i	components of $\vec{y}$
Z	third component of the helicopter position in the observer frame (altitude)
$\vec{\Omega}_{i/i-1}$	rotation speed of frame i into frame $i-1$
ρ	density of the fluid
ρ'	density perturbation, $\rho - \rho_0$
τ	source time

Subscripts

L	loading noise component
ret	quantity evaluated at retarded time $\tau = t - r/c$
T	thickness noise component
0	fluid variable in quiescent medium

References

[1] K.S. Brentner, Prediction of helicopter rotor discrete frequency noise, NASA TM 87721, 1986.
[2] J.M. Gallman, The validation and application of a rotor acoustic prediction computer program, NASA TM 101794, 1990.
[3] P. Spiegel, Prediction and analysis of the noise emitted by a main rotor of a helicopter in case of BVI, ONERA Publication No. 1996-1, 1996.
[4] K.S. Brentner, H.E. Jones, Noise prediction for maneuvering rotorcraft, AIAA Paper 2000-2031, 2000.
[5] J.E. Ffowcs Williams, D.L. Hawkings, Sound generation by turbulence and surfaces in arbitrary motion, *Philosophical Transactions of the Royal Society, London* A264 (1151) (1969) 321–342.
[6] K.S. Brentner, F. Farassat, Analytical comparison of the acoustic analogy and Kirchoff formulation for moving surfaces, *American Institute of Aeronautics and Astronautics Journal* 36 (8) (1998) 1379–1386.
[7] F. Farassat, G.P. Succi, The prediction of helicopter discrete frequency noise, *Vertica* 7 (4) (1983) 309–320.
[8] P. Agati, Y. Brémont, G. Delville, *Mécanique du Solide: Applications Industrielles*, Dunod, Paris, 1998 (in French).
[9] H. Rahnejat, *Multi-body Dynamics*: *Vehicle*, *Machines*, *and Mechanisms*, Professional Engineering Publishing (Imech), London, 1998 (see pg. 59, Relation 2.152).
[10] G. Perez, Investigation of the Influence of Maneuver on Rotorcraft Noise, MS Thesis, Department of Aerospace Engineering, The Pennsylvania State University, University Park, PA.
[11] K.S. Brentner, Numerical algorithms for acoustic integrals with examples for rotor noise prediction, *American Institute of Aeronautics and Astronautics Journal* 35 (4) (1997) 625–630.
[12] Y. Özyörük, L.N. Long, A Navier–Stokes/Kirchhoff method for noise radiation from ducted fans, AIAA Paper 94-0462, 1994.
[13] J.G. Leishman, Aeroacoustics of 2-D and 3-D blade vortex interaction using the indicial method, *American Helicopter Society 52nd Annual Forum Proceedings*, *Washington DC*, *June* 4–6, 1996.
[14] G.A. Brès, Modeling the Noise of Arbitrary Maneuvering Rotorcraft: Analysis and Implementation of the PSU-WOPWOP Noise Prediction Code, MS Thesis, Department of Aerospace Engineering, The Pennsylvania State University, University Park, PA, 2002.
[15] K.S. Brentner, C.L. Burley, M.A. Marcolini, Sensitivity of acoustic prediction to variation of input parameters, *Journal of the American Helicopter Society* 39 (3) (1994) 43–52.
[16] W. Johnson, Comprehensive analytical model of rotorcraft aerodynamics and dynamics, Vols. I–VII, Johnson Aeronautics Report, 1998.