Computers & Fluids xxx (2010) xxx–xxx

Contents lists available at ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

A hybrid algorithm for far-field noise minimization

Markus P. Rumpfkeil *, David W. Zingg

University of Toronto, Institute for Aerospace Studies, Toronto, ON, Canada M3H 5T6

ARTICLE INFO

Article history:
Received 8 July 2009
Received in revised form 18 April 2010
Accepted 11 May 2010
Available online xxxx

Keywords:
Unsteady adjoint
Unsteady optimization
Noise prediction
Far-field noise minimization
Ffowcs Williams and Hawkings wave propagation
Remote inverse shape design

ABSTRACT

A general unsteady adjoint formulation is applied to a hybrid acoustic prediction algorithm to provide an efficient far-field noise minimization algorithm. Two-dimensional unsteady Navier–Stokes (NS) computations for calculating the properties of acoustic sources are combined with the Ffowcs Williams and Hawkings (FW–H) wave propagation formulation to calculate the resulting far-field noise. Two different time-marching methods, namely an implicit multi-stage and an implicit multi-step method, are used for time discretization. The hybrid NS/FW–H solver is verified by comparison to an analytical solution and a Navier–Stokes solution. A discrete-adjoint Newton–Krylov algorithm is used to enable gradient-based shape optimization to minimize far-field noise computed using the hybrid solver. Objective functions considered include remote inverse shape designs for verification as well as the far-field pressure fluctuations for a blunt trailing edge airfoil in an unsteady turbulent flow environment. The examples presented demonstrate that the combination of a discrete-adjoint Newton–Krylov algorithm with a hybrid NS/FW–H far-field noise prediction method can be an efficient design tool for reducing aerodynamically generated noise.

1. Introduction and motivation

Airframe-generated noise is an important component of the total noise radiated from commercial aircraft, especially during aircraft approach and landing, when engines operate at reduced thrust, and airframe components (such as high-lift devices) are in the deployed state [1–4]. Future Federal Aviation Administration noise regulations, the projected growth in air travel, and the increase in population density near airports will require future civil aircraft to be substantially quieter than the current ones. Consequently, the attempt to understand and reduce airframe noise has become an important research topic [5].

A typical approach to tackle airframe-generated noise computations is to calculate the computational fluid dynamics (CFD) solution on a reasonable mesh that does not extend too far from the aircraft. A near-field plane or surface within the computational mesh can then serve as an interface between the CFD solution and a wave propagation program [6–8]. Such a program is able to model the wave propagation and to calculate the pressure fluctuations at a user specified ground plane which can then be used as a measure of the airframe-generated noise (see Fig. 1). Such hybrid acoustic predictions have been suggested about thirty years ago [6]. Several prediction methodologies for far-field signals based on near-field inputs with a solid physical and mathematical basis are currently available. The most popular among them are the Kirchhoff approach [9,10] and the Ffowcs Williams and Hawkings (FW–H) approach [11], which is based on the Lighthill acoustic analogy [12]. The FW–H equation is analytically superior for aeroacoustics because it is based upon the conservation laws of fluid mechanics rather than the wave equation [13] which means that the FW–H equation is still valid if the near-field surface is in the non-linear flow region. The Kirchhoff approach can lead to substantial errors if this surface is not positioned in the linear region [13,14]. The main difficulty in solving the FW–H equation in two dimensions is the semi-infinite time integral that arises when using the appropriate two-dimensional Green function in the time-domain [15]. This "tail effect" requires an infinitely long time to account for all contributions of the sources and is thus infeasible. However, the FW–H equation can be transformed into the frequency-domain to avoid this problem [16,15] and this approach is the wave propagation formulation of choice in this work.

Examples of hybrid predictions with the unsteady Reynolds-averaged Navier–Stokes (URANS) equations can be found in the literature for a supersonic cavity flow [17], radiated sound for a circular cylinder [18], and more recently for a slat trailing-edge flow [19–22]. Singer et al. [20] compared two- and three-dimensional hybrid NS/FW–H solutions for slat far-field noise and demonstrated the usefulness of the two-dimensional results, which gave the correct features of the radiated sound, but overpredicted the amplitude. Choudhari et al. [23] also stated that the two-dimensional approach should be sufficient for the purpose of understanding the basic noise generation mechanisms and

* Corresponding author. Tel.: +1 307 760 4694; fax: +1 307 766 2695.
E-mail address: markus@oddjob.utias.utoronto.ca (M.P. Rumpfkeil).
URL: http://goldfinger.utias.utoronto.ca/~dwz/ (D.W. Zingg).

0045-7930/$ - see front matter Crown Copyright © 2010 Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2010.05.006

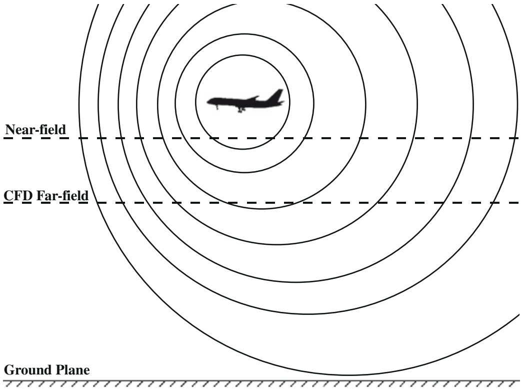

Fig. 1. Schematic of the propagation of the aircraft pressure signature.

determining the effect of Mach number and angle of attack changes. This implies that two-dimensional hybrid NS/FW–H simulations can be used to find trends, even though they do not represent all of the underlying physics exactly. Eventually one should use three-dimensional computations and large eddy simulations (LES) or at least a hybrid NS/LES approach to capture the near-field physics more exactly. However, for the current state-of-the-art computers, such simulations are very time-consuming and hence are not yet commonly used in the context of optimization. Since unsteady two-dimensional Navier–Stokes simulations are much cheaper and do a reasonable job of modeling vortex shedding caused by trailing-edge bluntness [24], which is the investigated test case here, we use them as our physical model in order to demonstrate and evaluate our algorithm. The 2D URANS model is used as an example, but the algorithm extends directly to three dimensions and LES or hybrid NS/LES.

The objective of this paper is to present and evaluate an efficient adjoint-driven gradient-based optimization algorithm to minimize far-field noise computed using a two-dimensional hybrid NS/FW–H solver. We employ a general framework to derive a discrete adjoint method for the optimal control of unsteady flows previously developed by the authors [25,26] with a Newton–Krylov approach to optimization [27,28] which together yield a very efficient optimization algorithm. Other researchers have also developed the capability of doing unsteady optimization using the adjoint method [29–33]. The main contributions of this work are the first application of an adjoint method for gradient evaluation to a hybrid NS/FW–H solver and a higher-order implicit time-marching method.

The organization of this paper is as follows. In Section 2 the hybrid NS/FW–H solver is presented and verified. Section 3 gives an overview of our unsteady optimization procedure and Section 4 presents the computational results using the novel hybrid NS/FW–H optimization algorithm. Finally, Section 5 concludes this paper.

2. Hybrid NS/FW–H solver

The implementation of our two-dimensional hybrid NS/FW–H solver is described in detail in this section. Section 2.1 describes the Navier–Stokes solver, and the two-dimensional FW–H solver is presented in Section 2.2. The FW–H solver and the combined hybrid NS/FW–H solver are verified in Section 2.3.

2.1. Navier–Stokes solver

The unsteady compressible two-dimensional thin-layer Navier–Stokes equations are solved using a well-validated Newton–Krylov algorithm [34,28]. The algorithm has recently been extended to unsteady flows with published verification and validation [35,26,36]. The equations are solved in generalized coordinates using a spatial discretization based on ARC2D [37], which consists of second-order centered-difference operators with second- and fourth-difference scalar artificial dissipation. For turbulent flows, eddy viscosity is computed using the Spalart–Allmaras turbulence model [38]. Second-order accurate time-marching is achieved through the backwards difference (BDF2) method covering a time interval $[0, T]$. The time-dependent flow solution Q^n for $n = 1, \ldots, N$ is then implicitly defined via

$$\begin{aligned}\mathcal{R}^n(Q^n, Q^{n-1}, Q^{n-2}, Y) &:= \frac{dQ^n}{dt} + R(Q^n, Y) \\ &= \frac{3Q^n - 4Q^{n-1} + Q^{n-2}}{2\Delta t} + R(Q^n, Y) = 0, \qquad (1)\end{aligned}$$

where $R = R(Q^n, Y)$ contains the spatially discretized convective and viscous fluxes as well as the boundary conditions and turbulence model, and Y are design variables used in the optimization process. N can be calculated from the relation $T = N\Delta t$, where Δt is the chosen time discretization step. For higher order accuracy in time we use the explicit first stage, single diagonal coefficient, diagonally implicit Runge–Kutta scheme of fourth-order (ESDIRK4) developed by Bijl, Carpenter and Vatsa [39]. It is implemented in our codes as a five-stage process [35]:

$$\mathcal{R}^n_k(Q^n_k, \ldots, Q^n_2, Q^{n-1}) := \frac{Q^n_k - Q^{n-1}}{a_{kk}\Delta t} + R(Q^n_k) + \frac{1}{a_{kk}} \sum_{j=1}^{k-1} a_{kj} R(Q^n_j) = 0$$
$$\text{for } k = 2, \ldots, 6. \qquad (2)$$

Here Q^n_k is the solution for the next time step n at stage k, given the solutions at the previous time level Q^{n-1} and previous stages Q^n_j with $j = 1, \ldots, k - 1$. The sixth and last stage gives the solution at the new time level, that is $Q^n := Q^n_6$. The terms a_{kj} are the Butcher coefficients for the scheme, which are given in Appendix A.

We use an inexact Newton strategy [35,34] to drive the discretized unsteady flow residual $\mathcal{R}^n$ to 10^{-10} at each time step n (or stage k for ESDIRK4). The main components of this strategy include the matrix-free generalized minimum residual (GMRES) method [40] and an incomplete lower-upper factorization [41] ILU(k) right preconditioner with a fill level of $k = 4$ to inexactly solve the linear system which results from applying Newton's method to the root finding problem for the unsteady flow residuals. The preconditioner is based on a first-order approximation of the flow Jacobian matrix, and the matrix–vector products required at each GMRES iteration are formed in a Jacobian-free manner with first-order finite differences.

2.2. FW–H solver

With the unsteady CFD solution in the near-field surface points $\mathbf{y}_s = (y_{s_1}, y_{s_2})$ (given by $f(\mathbf{y}_s) = 0$ such that $\frac{\partial f}{\partial y_i} = n_i$ is the unit normal vector that points into the fluid) as an input, the FW–H equation after a Galilean transformation can be written as [15]:

$$\begin{aligned}&\left\{\frac{\partial^2}{\partial t^2} + U_i U_j \frac{\partial^2}{\partial y_i \partial y_j} + 2U_j \frac{\partial^2}{\partial y_j \partial t} - a_\infty^2 \frac{\partial^2}{\partial y_i^2}\right\}[\rho' H(f)] \\ &\quad = \frac{\partial}{\partial t}[\mathcal{Q}\delta(f)] - \frac{\partial}{\partial y_i}[\mathcal{F}_i \delta(f)] + \frac{\partial^2}{\partial y_i \partial y_j}[T_{ij} H(f)]. \qquad (3)\end{aligned}$$

Here, the monopole term $\mathcal{Q}$, dipole term $\mathcal{F}_i$, and quadrupole term or Lighthill stress tensor T_{ij} are defined as

$$\mathcal{Q}(\mathbf{y}_s, t) = \rho u_j n_j, \qquad (4)$$
$$\mathcal{F}_i(\mathbf{y}_s, t) = [\rho(u_i - 2U_i)u_j + p\delta_{ij} - \tau_{ij}]n_j, \qquad (5)$$
$$T_{ij}(\mathbf{y}_s, t) = \rho(u_i - U_i)(u_j - U_j) + (p - a_\infty^2 \rho)\delta_{ij}, \qquad (6)$$

where $\rho = \rho_\infty + \rho'$, $u_i = U_i + u_i'$ and $p = p_\infty + p'$ are the total density, velocity and pressure, respectively. Free-stream quantities are indicated by the subscript ∞, U_i are the components of the uniform mean velocity, and a prime denotes a perturbation from the mean. The Cartesian coordinates and time are y_i and t, respectively, δ_{ij} is the Kronecker delta, $H(f)$ is the Heaviside function, and repeated indices follow the usual Einstein summation convention.

After a Fourier transformation of Eq. (3) and some simplifications, the far-field pressure fluctuations in the frequency-domain at an observer position $\mathbf{y}_o = (y_{o_1}, y_{o_2})$ for $M < 1$ can be calculated from [15]:

$$p'(\mathbf{x}_o,\omega) = -\oint_{f=0} i\omega\mathscr{Q}(\mathbf{y}_s,\omega)G(\mathbf{y}_o,\mathbf{y}_s,\omega)dl - \oint_{f=0}\mathscr{F}_j(\mathbf{y}_s,\omega)\frac{\partial G(\mathbf{y}_o,\mathbf{y}_s,\omega)}{\partial y_j}dl - \int_{f>0}T_{jk}(\mathbf{y}_s,\omega)\frac{\partial^2 G(\mathbf{y}_o,\mathbf{y}_s,\omega)}{\partial y_j\partial y_k}d\mathbf{y}, \qquad (7)$$

with the two-dimensional free-space Green function given by

$$G(\mathbf{y}_o,\mathbf{y}_s,\omega) = \frac{i}{4\beta}\exp\left(iM_\infty k r_1/\beta^2\right)\cdot H_0^{(2)}\left(\frac{k}{\beta^2}\sqrt{r_1^2+\beta^2 r_2^2}\right), \qquad (8)$$

where

$$r_1 = (y_{o_1} - y_{s_1})\cos\theta + (y_{o_2} - y_{s_2})\sin\theta, \qquad (9)$$

$$r_2 = -(y_{o_1} - y_{s_1})\sin\theta + (y_{o_2} - y_{s_2})\cos\theta. \qquad (10)$$

The angle θ is defined via $\tan\theta = U_2/U_1$, $H_0^{(2)}$ is the Hankel function of the second kind of order zero, $k = \omega/a_\infty$ is the wavenumber, $M_\infty = \sqrt{U_1^2 + U_2^2}/a_\infty$ and $\beta = \sqrt{1 - M_\infty^2}$ is the Prandtl–Glauert factor. The quadrupole term is neglected due to its small contributions relative to the other terms for the type of flows investigated in this paper. Furthermore, we let near-field surface points $\mathbf{y}_s$ coincide with grid nodes and apply a fast Fourier transformation (FFT) to the monopole and dipole terms. The Green function and its derivatives are calculated analytically and the line integrals are computed using the trapezoidal rule.

2.3. Noise prediction verification

In order to verify the program code for solving the two-dimensional FW–H equation in the frequency-domain, two cases are presented. In the first case, the results for the far-field pressure fluctuations are compared to a well-known analytical solution. In the second case, a direct comparison between the FW–H output and that obtained from a CFD simulation is performed to gauge the validity of the formulation for airframe-generated noise.

2.3.1. Monopole in uniform flow

The verification case considered is the acoustic field from a monopole line source. Greschner et al. [42] and Lockard [15] used very similar verification cases. The complex velocity potential for a stationary monopole source placed at the origin in a uniform flow with velocity U_i and a flow angle $\tan\theta = U_2/U_1$ can be written as

$$\phi(y_1,y_2,t) = \mathscr{A}(y_1,y_2,t)H_0^{(2)}\left(\frac{k}{\beta^2}\bar{r}\right), \qquad (11)$$

with

$$\mathscr{A}(y_1,y_2,t) = A\frac{i}{4\beta}\exp(i\omega t + iM_\infty k\bar{x}/\beta^2), \qquad (12)$$

$$\bar{r} = \sqrt{\bar{x}^2 + \beta^2\bar{y}^2}, \qquad (13)$$

$$\bar{x} = y_1\cos\theta + y_2\sin\theta, \qquad (14)$$

$$\bar{y} = -y_1\sin\theta + y_2\cos\theta, \qquad (15)$$

where A is the amplitude of the source signal. The perturbation variables needed to calculate the monopole and dipole source terms are obtained from the real parts of [42]

$$p' = p - p_\infty = -\rho_\infty\left(\frac{\partial\phi}{\partial t} + U_1\frac{\partial\phi}{\partial y_1} + U_2\frac{\partial\phi}{\partial y_2}\right), \qquad (16)$$

$$u_i' = u_i - U_i = \frac{\partial\phi}{\partial y_i}, \qquad (17)$$

$$\rho' = \rho - \rho_\infty = \frac{p'}{a_\infty^2}. \qquad (18)$$

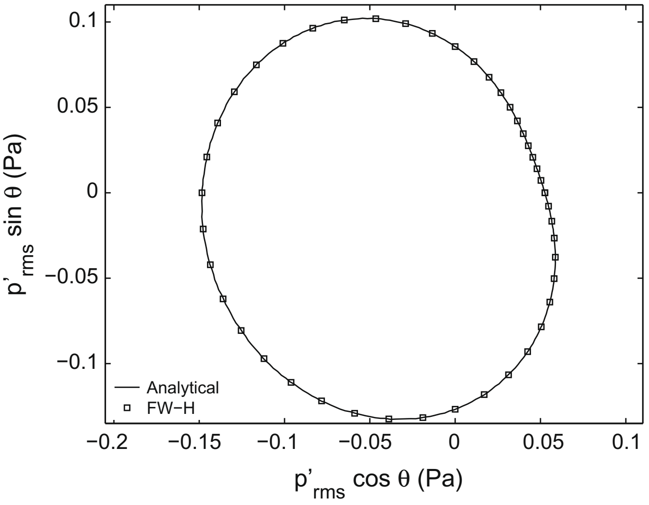

Fig. 2. Directivity comparison at $r = 500$ m.

The source terms are calculated from these flow variables evaluated over one period $T_P = 2\pi/\omega$ on the near-field integration surface, which is a circle with a radius of 2 m. One hundred uniformly spaced points on this circle are used as source locations. For this example,

$$A = 0.01\ \mathrm{m}^2/\mathrm{s},\quad M_\infty = 0.5,\quad \omega = 3000.0\ \mathrm{rad/s},$$
$$\theta = 20^\circ,\quad p_\infty = 1.00016\times 10^5\ \mathrm{Pa},\quad T_\infty = 300\ \mathrm{K}.$$

Fig. 2 compares the directivity from the FW–H calculation to the analytical solution in the far-field at $r = 500$ m. The agreement is excellent, demonstrating that the two-dimensional FW–H formulation is verified for problems with a uniform mean flow.

2.3.2. Airfoil in laminar flow

As a verification case, a direct comparison between the pressure fluctuations calculated via the FW–H approach and those obtained from a CFD simulation is performed at three distinct locations with increasing distance from an airfoil. The laminar flow over the single-element NACA 0012 airfoil with a Reynolds number of 800, a free stream Mach number of 0.2 and an angle of attack of 20° is considered. At these conditions the airfoil experiences vortex shedding. A very fine C-mesh with 848 × 395 nodes for high accuracy and a small non-dimensional time step of $\Delta t = 0.03$ are used with BDF2. After the flow solver has reached a periodic steady state, 1800 time steps are taken, which cover about six vortex shedding cycles, and the solution is recorded. The fictitious or permeable FW–H integration surface consists of one of the streamwise grid lines located at about 0.015c away from the airfoil, and the surface is closed at the trailing edge with one of the existing vertical grid lines.

The extracted CFD pressure fluctuations and those computed using the FW–H solver are plotted in Fig. 3 for the three probe locations. The wavelength of the dominant tone of this quasi point

Please cite this article in press as: Rumpfkeil MP, Zingg DW. A hybrid algorithm for far-field noise minimization. Comput Fluids (2010), doi:10.1016/j.compfluid.2010.05.006

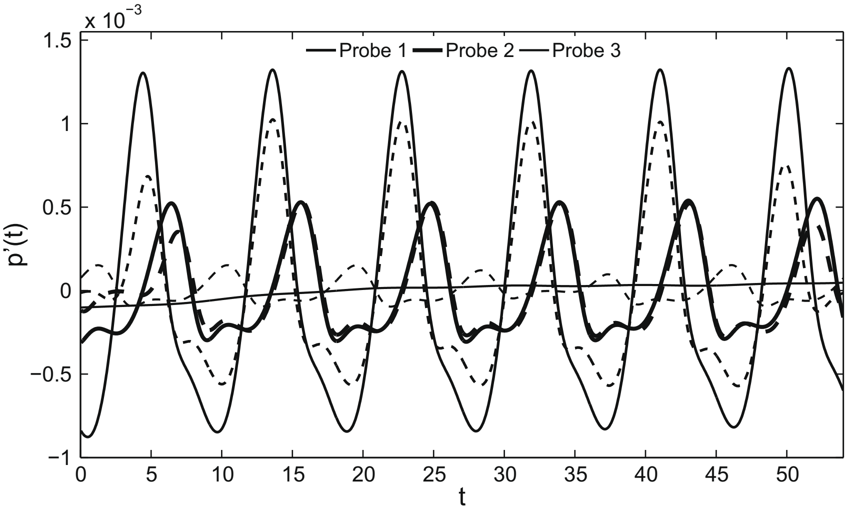

Fig. 3. Comparison of pressure fluctuations calculated by CFD (solid) and FW–H (dashed) at three probe locations.

source is given in non-dimensional units (chord lengths) by $\lambda = a_\infty \cdot T_P \approx 1 \cdot 9.2 = 9.2$. In two dimensions one expects that the sound intensity, which itself is proportional to the square of the sound pressure, is inversely proportional to the distance of an acoustic point source. This distance law is almost perfectly fulfilled by the pressure fluctuations which are calculated with the FW–H approach.

Comparing the pressure fluctuations probe location by probe location, one can make the following observations: At probe location 2, which is about 2c below the airfoil, the two pressure records are almost identical, except for the beginning and end of the data, where a window function [15] (required to make the input data periodic) tarnishes the result from the FW–H approach.

The agreement in the first probe location (less than 1c below) is also fairly good, except for the underprediction of the amplitude by the FW–H calculation. The very strong wake passing through the FW–H surface can cause errors [43] and might explain this discrepancy. Another possible explanation is that the quality of the input data to the FWH prediction code is not accurate enough. The CFD results at probe location 3 (30c below) are basically useless due to the coarser grid this far away from the airfoil, and it is apparent that an accurate propagation of the pressure signatures to the far-field for moderate cost is only achievable with an acoustic wave propagation code.

3. Outline of unsteady optimization algorithm

Our unsteady optimization procedure is as follows. First, the governing flow equations are solved as described in Section 2.1. Once the time-dependent flow solution Q^n has been calculated and stored for $n = 1,\ldots,N$, a discrete cost function J is evaluated

$$J = \sum_{n=1}^{N} I^n(Q^n, Y), \tag{19}$$

where the function $I^n = I^n(Q^n, Y)$ depends on Q^n and design variables Y. In our case the evaluation of the cost function is fairly involved since Q^n is the input to the FW–H solver (see Section 2.2) which Fourier transforms computed monopole and dipole terms, calculates the line integrals given by Eq. (7) and inverse Fourier transforms the resulting pressure fluctuations in the observer locations.

Next, the appropriate adjoint equations (given in Appendix A for both time-marching methods) are solved by going backward in time. The Bi-CGSTAB algorithm [44] is used to solve the linear adjoint equations with an absolute convergence tolerance of 10^{-6}, and right preconditioning with ILU(5) is applied to accelerate convergence. The chosen convergence tolerance is a good balance between avoiding error propagation and spending too much computational time. We found Bi-CGSTAB to be up to 50% faster in solving the unsteady adjoint equations than GMRES which is more efficient for steady-state adjoint problems [27,28]. The reason for this is that the transpose of the unsteady flow Jacobian $(\nabla_{Q^n}\mathscr{R}^n)^T$ is more diagonally dominant than the transpose of the steady flow Jacobian $(\nabla_Q R)^T$ due to the extra terms on the diagonal, which makes this matrix more suited for the use of Bi-CGSTAB. Note that since the FW–H equation is used as part of the cost function in the optimization framework, the derivative of the far-field pressure fluctuations with respect to the near-field input flow variables is needed explicitly. This differentiation is done by hand using the chain rule and is somewhat cumbersome. However, this could be alleviated using automatic differentiation tools (ADOL-C, ADIFOR, TAPENADE, etc.).

The fourth step is to calculate the gradient of J with respect to the design variables Y (details are given in Appendix A). The shape design variables Y are based on a cubic B-spline parametrization of the airfoils; we use the vertical coordinates of selected B-spline control points as design variables [27]. It takes about 2–3 times the computational time of an unsteady flow solution to calculate the corresponding gradient. Furthermore, the flow data from the forward time calculation must either be stored or recalculated on the fly [45]. For our 2D simulations we can easily afford to store the entire flow-field with negligible impact on the computational time due to disk access.

Lastly, a quasi-Newton BFGS optimizer [46,47] gives us new design variable values, the airfoil geometry is updated accordingly, and the grid is modified using an algebraic grid movement algorithm [28]. In order to avoid large shape changes we impose box constraints on all design variable values. The entire process is repeated until the gradient is reduced by a few orders of magnitude which implies that the cost function is close to a minimum.

At this time a comment is in order regarding the widely held notion that optimization of unsteady flows using adjoint methods is too expensive. The efficiency of the overall procedure depends on the cost of the flow solver, the cost of the gradient evaluation, and the number of optimization iterations required. Using the adjoint method the cost of computing the gradient is of the order of the cost of the flow solve. For our particular implementation the overall computational cost is roughly $3.5 \cdot N_{oi} \cdot T_{uf}$, where N_{oi} is the required number of optimization iterations and T_{uf} is the computational time for one unsteady flow solution and cost function evaluation. We will demonstrate in the results section that the value of N_{oi} is comparable to steady state problems with a similar number of design variables. Thus, the ratio of the cost of unsteady adjoint optimization to unsteady flow solution is quite similar to the steady case. In conclusion, the only factor left which makes unsteady optimization more expensive than steady optimization is T_{uf}, the computational time required to solve the flow and evaluate the cost function. For the time being, restrictions to two dimensions, URANS models and/or non-linear frequency-domain methods [48] for unsteady optimizations are warranted. However, these limitations will most likely be eased in the near future.

4. Results

4.1. Verification: remote inverse shape design in laminar flow

For verification purposes, several remote inverse shape design problems are presented with a discrete cost function given by

$$J = \frac{1}{2}\Delta t \sum_{n=N_c+1}^{N} \left(p_{obs}^n - p_{obs}^{*n}\right)^2. \tag{20}$$

Here, p_{obs}^n is the pressure at some far-field observer location at time step n obtained from a current airfoil shape, and p_{obs}^{*n} is the target pressure at the same observer location and time step obtained from the target airfoil shape. N_c is the number of time steps used in the adjusting period where the cost function is not yet computed. The target shape is given through a perturbation in four shape design variables of the initial NACA 0012 airfoil; both shapes are shown in Fig. 4. Only four design variables are used to keep the problem simple and to be able to compare the adjoint gradient with a finite-differenced one in order to verify the accuracy of the gradient calculation.

The unsteady flow conditions for this test case are exactly the same as the ones used in the verification of the acoustic propagation code in Section 2.3.2, namely a Reynolds number of 800, a free-stream Mach number of 0.2, and an angle of attack of 20°. A coarser mesh with only about 35,000 nodes is used to reduce the computational costs. Nonetheless, as displayed in Fig. 5, the comparisons of pressure fluctuations calculated by CFD and FW–H for the initial NACA 0012 show a good agreement at a point about two chord lengths below the trailing edge for both time-marching methods. However, as compared to the second probe location in Fig. 3 the magnitude is off by a factor of two which indicates that the solution is not grid converged, but it is still good enough to verify our overall optimization procedure. We will reanalyze the optimized shapes for the more realistic optimizations in Section 4.3 using a much finer mesh to confirm that those optimizations did in fact generate a shape that substantially reduces the objective functions.

Fig. 6 shows the drag coefficients for the initial and target airfoils over time using a time step of $\Delta t = 0.05$ for BDF2 and $\Delta t = 0.5$ for ESDIRK4. One can see the adjustment period for the target airfoil. In order to reduce the computational costs in the actual optimization runs, a bigger time step of $\Delta t_c = 0.1$ is utilized for the first $N_c = 200$ steps with BDF2, and $\Delta t_c = 0.5$ for the first $N_c = 40$ steps is used with ESDIRK4. Once the domain where the pressures are compared is reached, a smaller time step $\Delta t = 0.05$ is used for another 1200 steps with BDF2 ($\Delta t = 0.5$ for 120 steps with ESDIRK4), leading to $N = 1400$ ($N = 160$ with ESDIRK4) time

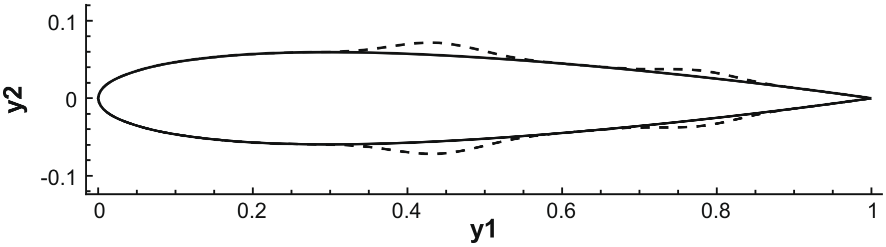

Fig. 4. The initial (solid) and target (dashed) airfoil shapes.

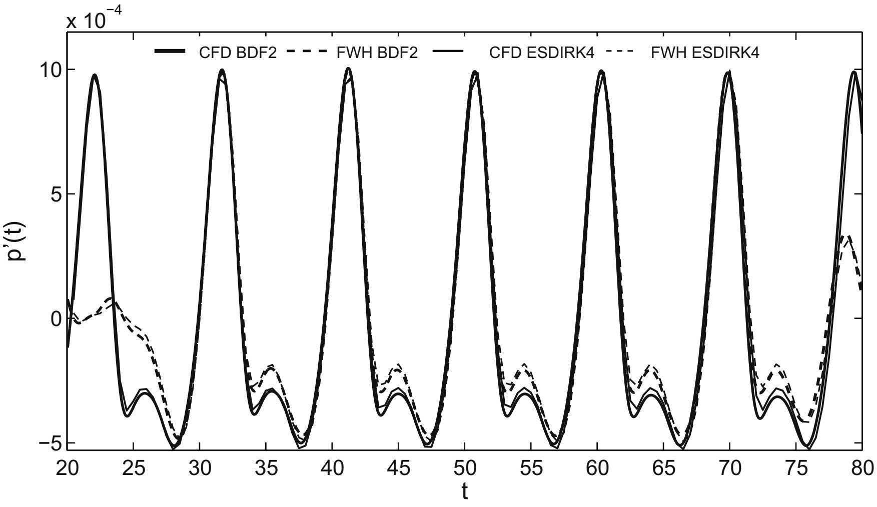

Fig. 5. Comparison of pressure fluctuations of the initial airfoil calculated by CFD (solid) and FW–H (dashed) about 2c below the trailing edge for two time-marching methods.

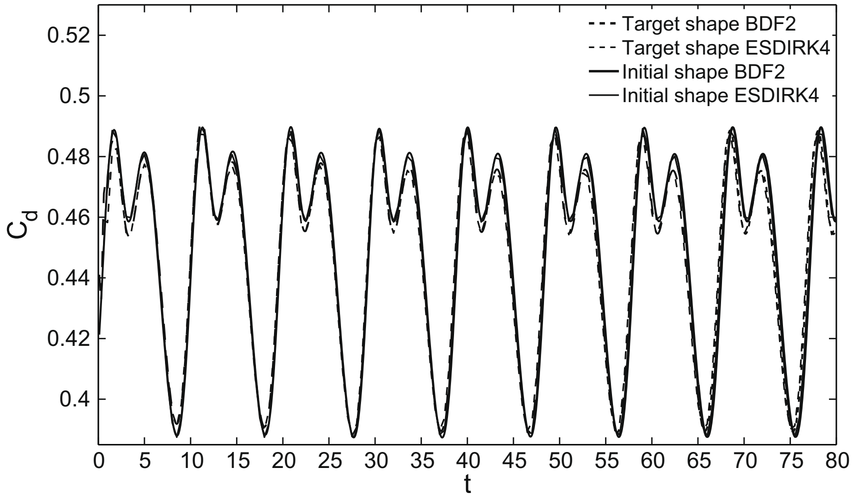

Fig. 6. Drag coefficients for the initial and target airfoil shapes.

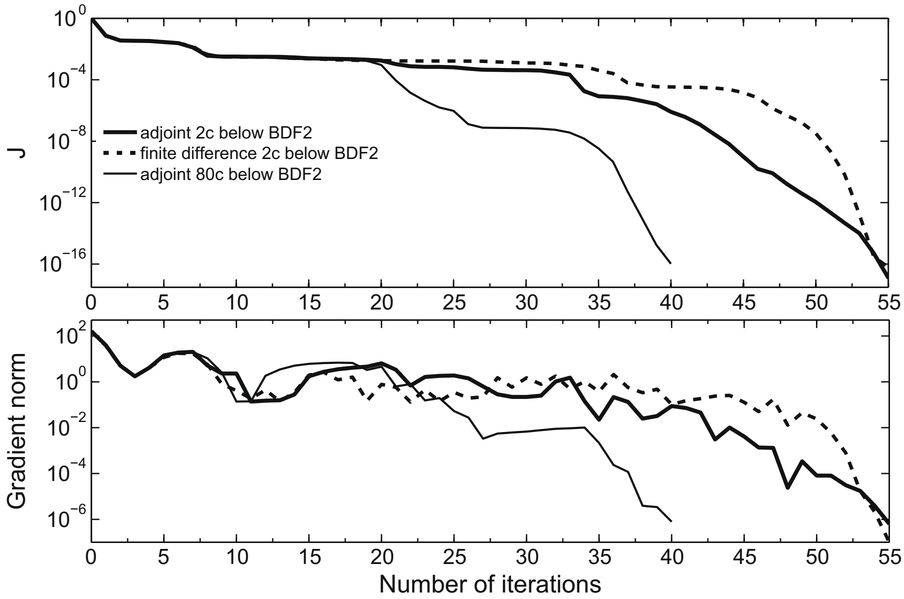

Fig. 7. Convergence histories of remote inverse shape design problems using the hybrid NS/FW–H algorithm with four design variables and BDF2 time-marching method.

steps in total for each flow solve covering a time interval of [0,80]. The corresponding adjoint equations for this situation are given in Appendix A.

The convergence histories of remote inverse shape design problems using the hybrid NS/FW–H optimization algorithm are presented in Figs. 7 and 8. The objective function is always scaled such that its initial value for either time-marching method is unity. The convergence history for an observer location eighty chord lengths below the leading edge using only the adjoint approach is shown in the figure. The adjoint approach is also compared to a second-order central finite-difference approach with a step size of 10^{-7} for a location that is about two chord lengths below the trailing edge. One can see that the objective functions are driven to small values in about forty to fifty design iterations, and that the two approaches show a reasonable agreement for both time-marching methods. The finite-difference approach for ESDIRK4 (dashed black line) does not fully converge, and the scaled objective function value stalls at about 10^{-8}. The reason for this is an inaccurate gradient due to error cancellation for such small values of the objective function in combination with the more involved time-marching method. The gradient at the first design iteration with BDF2 using the adjoint method (*ad*) yields

$$\left(\frac{\partial J}{\partial Y}\right)_{ad} = (-48.09, -162.26, -55.78, -36.68),$$

which is in good agreement with the gradient calculated via the finite-difference method (*fd*):

$$\left(\frac{\partial J}{\partial Y}\right)_{fd} = (-52.83, -160.56, -56.68, -36.43).$$

Using ESDIRK4, the same gradient is given by

$$\left(\frac{\partial J}{\partial Y}\right)_{ad} = (-48.37, -166.64, -57.89, -37.76),$$

which also agrees well with the finite-difference gradient:

$$\left(\frac{\partial J}{\partial Y}\right)_{fd} = (-53.11, -164.44, -58.90, -37.60).$$

Please cite this article in press as: Rumpfkeil MP, Zingg DW. A hybrid algorithm for far-field noise minimization. Comput Fluids (2010), doi:10.1016/j.compfluid.2010.05.006

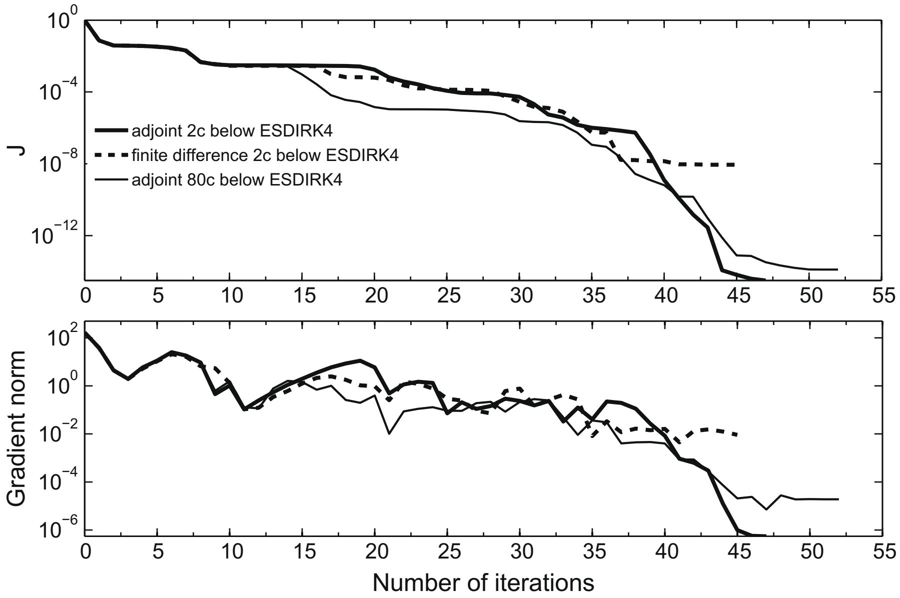

Fig. 8. Convergence histories of remote inverse shape design problems using the hybrid NS/FW–H algorithm with four design variables and ESDIRK4 time-marching method.

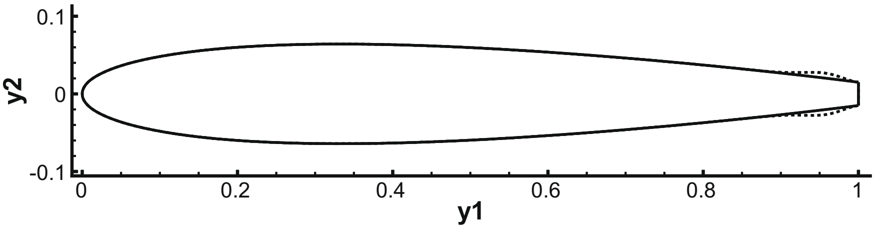

Fig. 9. The initial (solid) and target (dashed) airfoil shapes.

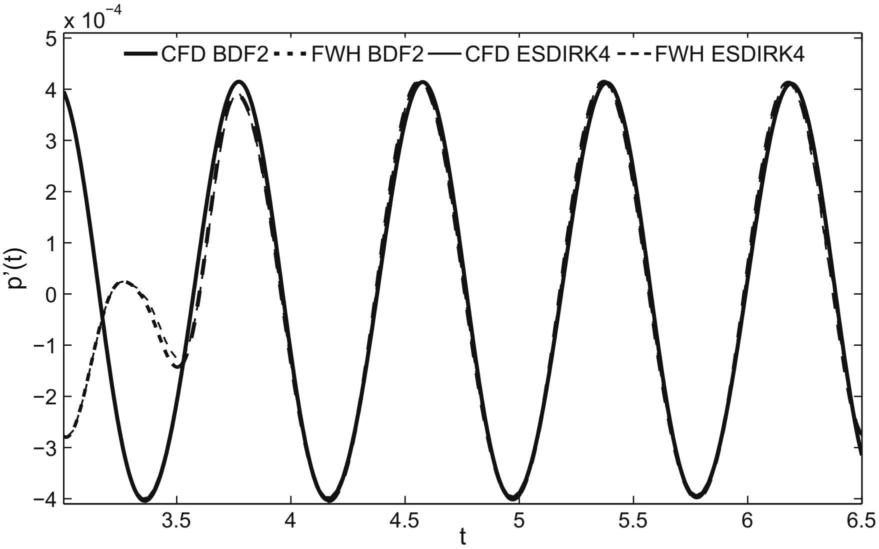

Fig. 10. Comparison of pressure fluctuations calculated by CFD (solid) and FW–H (dashed) about $\frac{1}{3}$c below the trailing edge of the initial airfoil for two time-marching methods.

Note that both gradients have some error. The maximum difference of about 10% in the gradients arises from the different error propagation in the flow and adjoint solutions (non-linear vs. linear) combined with the fact that in the adjoint approach the objective function and residual sensitivities with respect to the design variables Y are evaluated using finite differences. This gradient accuracy is definitely good enough for an optimization procedure.

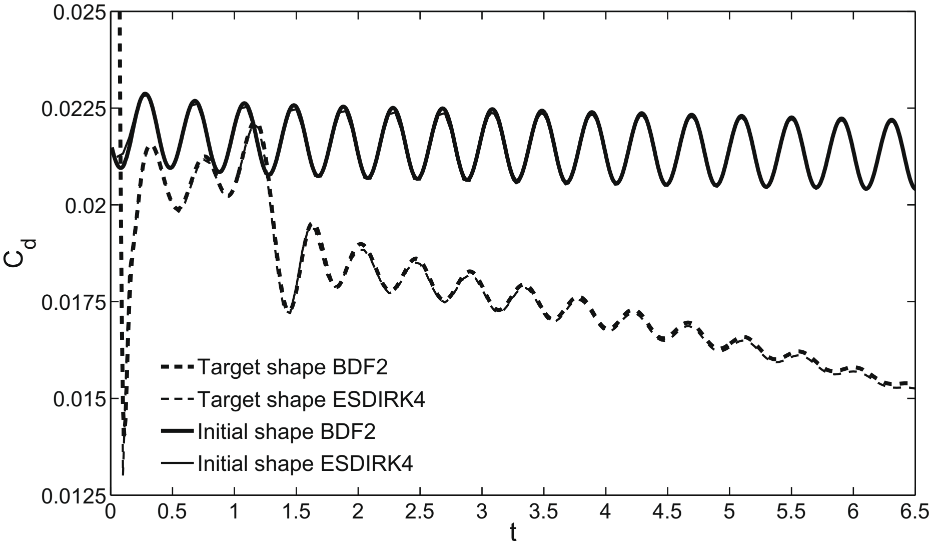

Fig. 11. Drag coefficients for the initial and target airfoil shapes.

Table 1
Time horizons for blunt trailing-edge flow.

	N_c	Δt_c	$N - N_c$	Δt	N
BDF2	300	0.01	700	0.005	1000
ESDIRK4	60	0.05	140	0.025	200

4.2. Verification: turbulent blunt trailing-edge flow

The shape of a NACA 0012 airfoil with a 0.03c thick blunt trailing edge in a turbulent flow is optimized in this section. The free-stream flow conditions are given by $M_\infty = 0.2$, $Re = 2 \times 10^6$, $\alpha = 0°$, and the mesh consists of about 36,000 nodes, which provides sufficient accuracy for our purpose, which is to demonstrate the applicability of the hybrid NS/FW–H optimization algorithm to far-field noise minimization problems. A remote inverse shape design problem is solved with the objective function given by Eq. (20) and a far-field observer located 40c below the leading edge. BDF2 and ESDIRK4 are again employed as time-marching methods, and only two shape design variables are used to enable a comparison between the adjoint gradient with a finite-differenced one and to thus verify the hybrid NS/FW–H optimization algorithm for turbulent flows. The initial and target airfoil shapes are shown in Fig. 9.

A comparison of pressure fluctuations calculated by CFD and FW–H at a location about $\frac{1}{3}$c below the trailing edge of the initial airfoil is displayed in Fig. 10 and shows good agreement for both time-marching methods. Fig. 11 shows the drag coefficients for the initial and target airfoil over time using a time step of $\Delta t = 0.005$ for BDF2 and $\Delta t = 0.025$ for ESDIRK4. Note that the target solution has not yet reached a periodic steady state, however, the computational costs are too high to cover the required time interval. The time horizons used in the remote inverse shape design for turbulent blunt trailing-edge flow are shown in Table 1. The chosen time step sizes lead to the same computational effort for a flow or adjoint solution for the two time-marching methods, since the computational cost for one time step using ESDIRK4 is roughly five times more expensive than using BDF2.

The convergence history of the remote inverse shape design problem for a turbulent blunt trailing-edge flow using the hybrid NS/FW–H optimization algorithm is presented in Fig. 12. The objective function is again always scaled such that its initial value for either time-marching method is unity. The adjoint approach is shown in comparison to a second-order central finite-difference approach with a step size of 10^{-5}. One can see that the objective functions are driven to small values in about ten design iterations and that the two approaches show a reasonable agreement for both time-marching methods, which implies that the adjoint approach for the gradient calculation is accurate.

In particular, the gradient at the first design iteration using the adjoint method (*ad*) yields

$$\left(\frac{\partial J}{\partial Y}\right)_{ad} = (-34.36, 35.11),$$

which is in good agreement with the gradient calculated via the finite-difference method (*fd*):

$$\left(\frac{\partial J}{\partial Y}\right)_{fd} = (-33.53, 34.18).$$

Similarly, the adjoint approach for ESDIRK4 at the first design iteration leads to

$$\left(\frac{\partial J}{\partial Y}\right)_{ad} = (-34.29, 35.14),$$

which agrees well with the finite-difference gradient:

$$\left(\frac{\partial J}{\partial Y}\right)_{fd} = (-33.71, 34.74).$$

Both approaches for ESDIRK4 do not fully converge, and the scaled objective function values stall at about 10^{-10}. The reason is again an inaccurate gradient due to error cancellation for such small values of the objective function in combination with the more complicated time-marching method.

4.3. Application to far-field noise minimization

After the verification of the hybrid NS/FW–H optimization algorithm for unsteady laminar and turbulent flows using two time-marching methods in the previous two sections, more practically relevant optimizations involving a turbulent blunt trailing-edge flow are considered here. Although this test case with periodic vortex shedding is not a key far-field noise generation mechanism, it is a representative problem that contains the important features of a

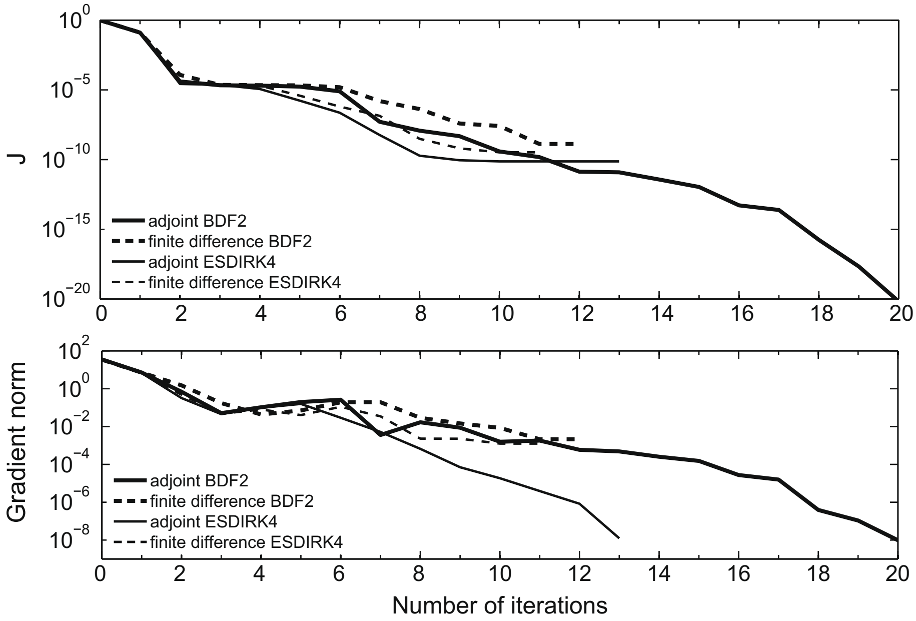

Fig. 12. Convergence history of the remote inverse shape design problem with two design variables.

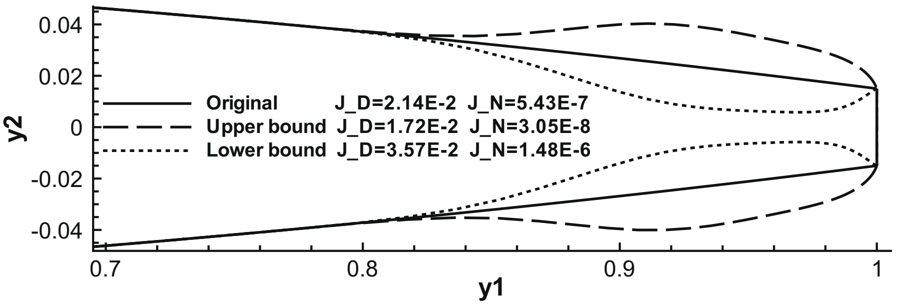

Fig. 13. Initial shapes for the turbulent blunt trailing-edge flow optimizations.

noise minimization problem. We consider two different objective functions:

1. Pressure fluctuation (noise) minimization

$$J_N = \sum_{n=N_c+1}^{N} \left(p_{obs}^n - \bar{p}_{obs}\right)^2 = \sum_{n=N_c+1}^{N} \left(p_{obs}'^n\right)^2. \qquad (21)$$

2. Mean drag minimization

$$J_D = \overline{C}_d = \frac{1}{N - N_c} \sum_{n=N_c+1}^{N} C_d^n, \qquad (22)$$

where $\bar{p}_{obs}$ is the mean pressure at the observer location, which is located 40c below the leading edge, and $p_{obs}'^n = p_{obs}^n - \bar{p}_{obs}$ is the pressure fluctuation in this observer location at time step n. The reasoning for the choice of these two objective functions is that we would like to see whether noise and drag improvements lead to identical, qualitatively similar or vastly different results for the shape of this airfoil. In Eq. (21) we consider only a single observer location, which could lead to a change in directivity rather than overall noise reduction. In general, a noise cost function should be defined as an integral over some region, and this is easily done, but for the present example the results show that a single observer location is sufficient.

Eight B-spline control points are used as shape design variables which are all located in the aft 15% of the chord length (four on the upper and four on the lower surface). The unsteady shape optimizations are started from three different initial shapes, which are shown in Fig. 13 together with their initial objective function values:

1. The initial airfoil (solid).
2. The airfoil that results from setting all eight design variables to their specified upper bounds (dashed).
3. The airfoil that results from setting all eight design variables to their specified lower bounds (dotted).

The flow conditions and the mesh are the same as given in Section 4.2, but only the BDF2 time-marching method is used here with the time horizon given in Table 1. Fig. 14 presents the final optimized airfoil shapes of the turbulent blunt trailing-edge flow optimizations together with their objective function values. For each objective function, all three initial shapes converge to the same respective final shapes. The mean drag value of the noise minimized airfoils is slightly higher than the mean drag value of the mean drag minimized airfoils and conversely, the noise objective function value J_N of the mean drag minimized airfoils is a factor of two higher than the one from the noise minimized airfoils.

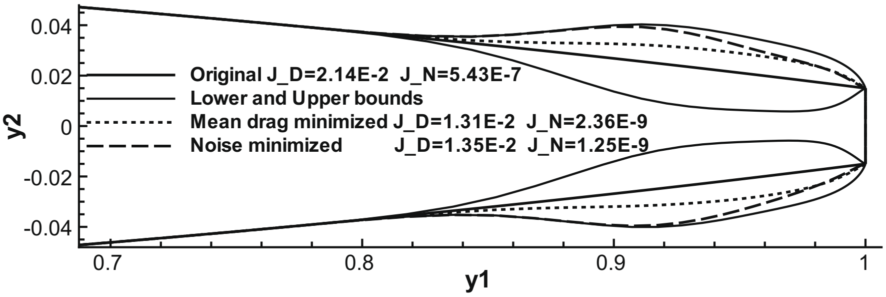

Fig. 14. Final improved airfoil shapes of the turbulent blunt trailing-edge flow optimizations.

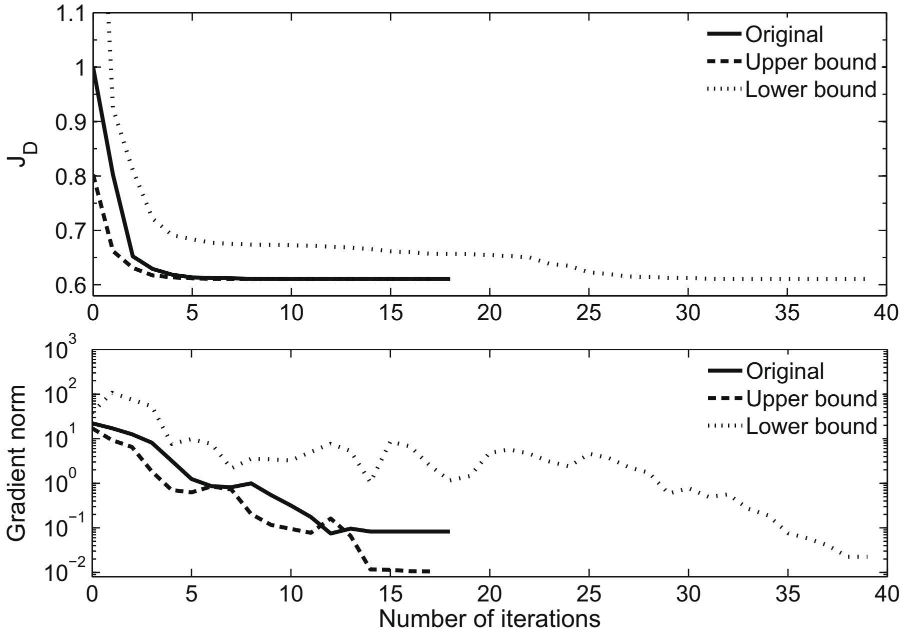

Fig. 15. Convergence histories of the mean drag minimizations for the turbulent blunt trailing-edge flow with eight design variables.

This shows that noise and drag improvements lead to qualitatively similar results to a first approximation, but they do not yield the same optimized shapes.

The convergence histories of the mean drag minimizations are displayed in Fig. 15. The objective function values are always scaled with the mean drag value of the original airfoil, $J_D = 2.14 \times 10^{-2}$, to make comparisons easier. Since all three initial shapes converge to the same final optimized shape, they have the same objective function value, which translates into a reduction in mean drag of about 39% from the original airfoil. The objective function value is mostly reduced in the first few iterations, and the improvements after that are only marginal. The gradient norms are reduced by 3–4 orders of magnitude indicating that the optimizer has converged to a minimum in each case.

The convergence histories of the noise minimizations in Fig. 16 show that this objective function is mainly reduced in the first five iterations and that the gradient norms are reduced by 2–3 orders of magnitude. The sum of the pressure fluctuations for the optimized shape is reduced to 0.23% of the initial value of the original airfoil $J_N = 5.43 \times 10^{-7}$, which is again used to scale the objective function values to ease comparisons. Starting from the lower bound leads to a failed line search in the first iteration because all gradients indicate that it would be beneficial to “slim” the airfoil even more, which is not permitted by the box constraints imposed on the design variables. This implies that there is a local minimum right on the constraint boundary, so one must start a sufficient distance from this boundary in order to converge to the global minimum.

The time histories of C_l and C_d for the original blunt trailing edge airfoil before and after the optimizations are shown in Fig. 17 using a time step of $\Delta t = 0.005$. One can clearly see the adjusting period for the improved airfoils in the time interval [0,3] before they reach their new somewhat periodic steady state. A reduced mean drag and constant mean lift for both optimized airfoils is also visible, and both objective functions lead to reduced oscillation amplitudes in both lift and drag. This implies that the noise in the entire flow-field has been reduced although we only use a single observer location where we minimize the pressure fluctuations.

We also reanalyzed the performance of the original and optimized airfoil shapes on a mesh with about 180,000 nodes. The lift and drag histories are very similar to the ones shown in Fig. 17. The mean drag value on this finer mesh changes for the original airfoil from $J_D = 2.14 \times 10^{-2}$ to $J_D = 2.19 \times 10^{-2}$, for the mean drag minimized airfoil from 1.31×10^{-2} to 1.52×10^{-2}, and for the noise minimized airfoil from 1.35×10^{-2} to 1.58×10^{-2}. Similarly, the noise objective function value changes for the original airfoil from $J_N = 5.43 \times 10^{-7}$ to $J_N = 5.28 \times 10^{-7}$, for the mean drag minimized airfoil from 2.36×10^{-9} to 6.51×10^{-8}, and for the noise minimized airfoil from 1.25×10^{-9} to 1.72×10^{-8}. Thus, giving very

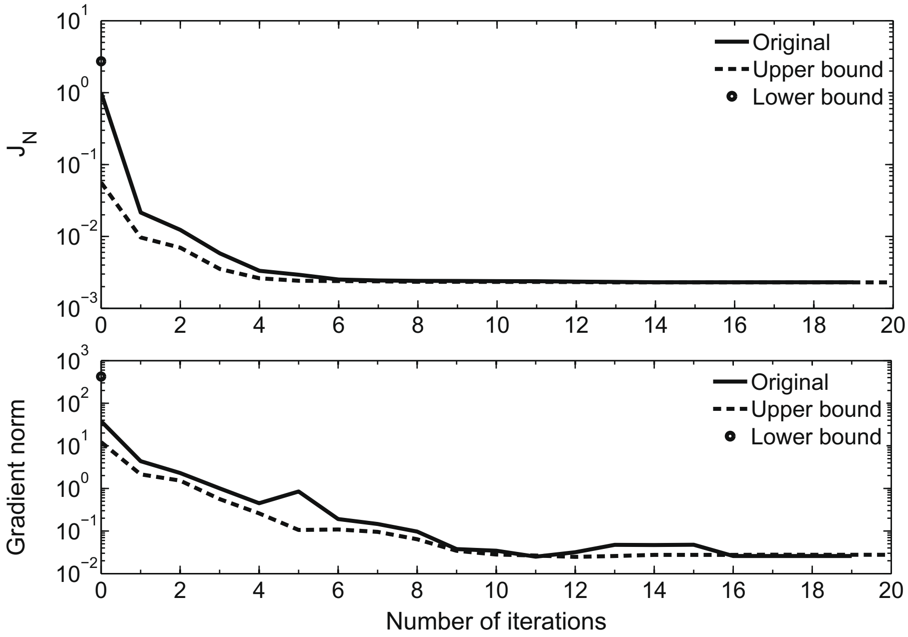

Fig. 16. Convergence histories of the noise minimizations for the turbulent blunt trailing-edge flow with eight design variables.

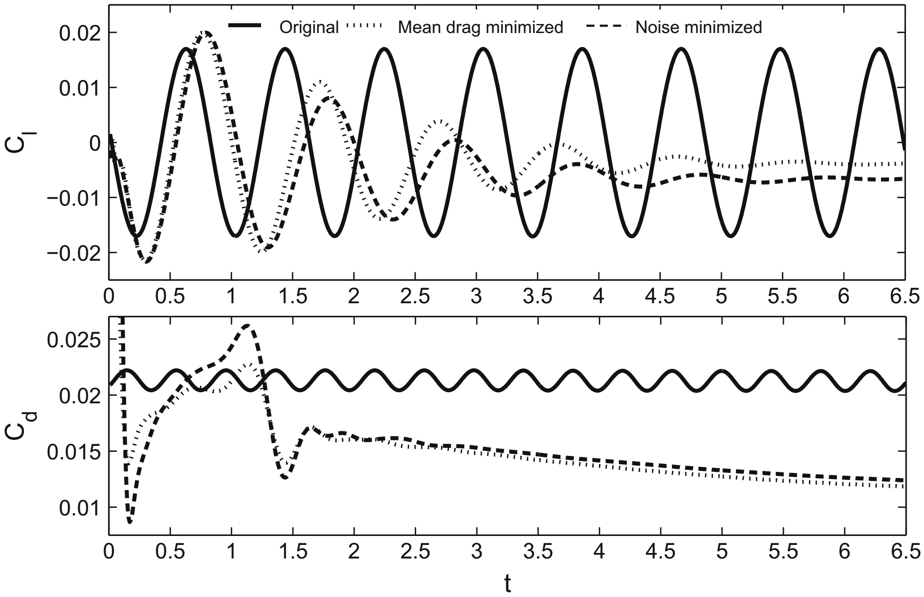

Fig. 17. Time histories of C_l and C_d before and after the optimizations vs. time ($\Delta t = 0.005$).

similar results or at least trends. As a last note, in a very recent paper Wang et al. [49] used a similar test case with a high-order discontinuous Galerkin discretization except that their case was inviscid and they minimized the near-field pressure fluctuations at the one-chord length circular locations from the center-point. They also obtained the somewhat bulgy shapes as an optimal solution.

5. Conclusions

The results presented in this paper show that the novel hybrid NS/FW–H optimization algorithm proposed, which uses a Newton–Krylov approach in combination with a discrete adjoint method, is effective and efficient for problems representative of practical applications. First we verified a hybrid NS/FW–H solver by comparison to an analytical solution and a Navier–Stokes solution. We also verified our hybrid NS/FW–H optimization algorithm by recovering far-field pressure fluctuations via remote inverse shape designs in unsteady laminar and turbulent flows. The unsteady turbulent flow around a blunt trailing edge airfoil example demonstrates that it is possible to minimize the pressure fluctuations at a given far-field observer position and thus that our hybrid NS/FW–H optimization algorithm is an efficient and practical design tool for reducing aerodynamically generated noise. The concepts presented in this paper also allow the use of more sophisticated, higher-order time-marching methods as demonstrated in Appendix A for ESDIRK4. Lastly, this optimization

Table 2
Butcher table for ESDIRK4 scheme [39].

a_{21}	$\frac{1}{4}$	a_{22}	$\frac{1}{4}$	0		0		0		0
a_{31}	$\frac{8611}{62{,}500}$	a_{32}	$\frac{-1743}{31{,}250}$	a_{33}	$\frac{1}{4}$	0		0		0
a_{41}	$\frac{50{,}12{,}029}{3{,}46{,}52{,}500}$	a_{42}	$\frac{-6{,}54{,}441}{29{,}22{,}500}$	a_{43}	$\frac{1{,}74{,}375}{3{,}88{,}108}$	a_{44}	$\frac{1}{4}$	0		0
a_{51}	$\frac{15{,}26{,}70{,}82{,}809}{1{,}55{,}37{,}62{,}65{,}600}$	a_{52}	$\frac{-7{,}14{,}43{,}401}{12{,}07{,}74{,}400}$	a_{53}	$\frac{73{,}08{,}78{,}875}{90{,}21{,}84{,}768}$	a_{54}	$\frac{22{,}85{,}395}{80{,}70{,}912}$	a_{55}	$\frac{1}{4}$	0
a_{61}	$\frac{82{,}889}{5{,}24{,}892}$	a_{62}	0	a_{63}	$\frac{15{,}625}{83{,}664}$	a_{64}	$\frac{69{,}875}{1{,}02{,}672}$	a_{65}	$\frac{-2260}{8211}$	$a_{66} = \frac{1}{4}$

algorithm can be extended to three dimensions as well as different flow solvers and physical models in a straightforward manner, although a parallel implementation is highly recommended for large-scale three-dimensional unsteady optimization problems.

Acknowledgments

The funding of the second author by the Natural Sciences and Engineering Research Council of Canada, the Mathematics of Information Technology and Complex Systems and the Canada Research Chairs program is gratefully acknowledged. We would also like to thank the reviewers for their very helpful comments and suggestions.

Appendix A

In this appendix, we derive the discrete adjoint equations in the form in which we use them to present our results. We begin with the equations for the BDF2 time-marching method (for more details see Rumpfkeil and Zingg [25,26]). We start our flow solve at $t = 0$ from a periodic steady state solution which implies that we know the flow solutions Q^0 and Q^{-1}. We want to "jump" over the adjusting period after a shape modification as quickly as possible thus taking a bigger time step Δt_c for N_c time steps. Once we reach the domain where we actually want to control the problem we use a smaller time step Δt for $N - N_c$ time steps. Thus we have a total of N time steps and to keep the second-order time accuracy, the time-dependent flow solution Q^n is implicitly defined via the following unsteady residuals:

$$\begin{aligned}
&\mathscr{R}^n(Q^n, Q^{n-1}, Q^{n-2}, Y) \\
&\quad := \frac{3Q^n - 4Q^{n-1} + Q^{n-2}}{2\Delta t_c} + R(Q^n, Y) = 0 \\
&\quad \text{for } n = 1, \ldots, N_c, \\
&\mathscr{R}^{N_c+1}(Q^{N_c+1}, Q^{N_c}, Q^{N_c-1}, Y) \\
&\quad := \frac{2\Delta t + \Delta t_c}{\Delta t(\Delta t + \Delta t_c)} Q^{N_c+1} - \frac{\Delta t + \Delta t_c}{\Delta t \Delta t_c} Q^{N_c} \\
&\qquad + \frac{\Delta t}{\Delta t_c(\Delta t + \Delta t_c)} Q^{N_c-1} + R(Q^{N_c+1}, Y) = 0, \\
&\mathscr{R}^n(Q^n, Q^{n-1}, Q^{n-2}, Y) \\
&\quad := \frac{3Q^n - 4Q^{n-1} + Q^{n-2}}{2\Delta t} + R(Q^n, Y) = 0 \\
&\quad \text{for } n = N_c + 2, \ldots, N.
\end{aligned}$$

The problem of minimizing the discrete objective function J as given by Eq. (19) subject to the convergence of the unsteady flow solve is equivalent to the unconstrained optimization problem of minimizing the Lagrangian function

$$\begin{aligned}
\mathscr{L}(Q^1, \ldots, Q^N, \psi^1, \ldots, \psi^N, Y) = &\sum_{n=N_c+1}^{N} I^n(Q^n, Y) \\
&+ \sum_{n=1}^{N} (\psi^n)^T \mathscr{R}^n(Q^n, Q^{n-1}, Q^{n-2}, Y),
\end{aligned}$$

with respect to $Q^1, \ldots, Q^N, \psi^1, \ldots, \psi^N$ and Y. The Lagrange multipliers ψ^n must be chosen such that $\nabla_{Q^n} \mathscr{L} = 0$ for $n = 1, \ldots, N$, which leads to

$$\begin{aligned}
\psi^N &= -\left(\left(\nabla_{Q^N} \mathscr{R}^N\right)^T\right)^{-1} \cdot \left[\left(\nabla_{Q^N} I^N\right)^T\right], \\
\psi^{N-1} &= -\left(\left(\nabla_{Q^{N-1}} \mathscr{R}^{N-1}\right)^T\right)^{-1} \cdot \left[\left(\nabla_{Q^{N-1}} I^{N-1}\right)^T + \left(\nabla_{Q^{N-1}} \mathscr{R}^N\right)^T \psi^N\right], \\
\psi^n &= -\left((\nabla_{Q^n} \mathscr{R}^n)^T\right)^{-1} \cdot \left[(\nabla_{Q^n} I^n)^T + (\nabla_{Q^n} \mathscr{R}^{n+1})^T \psi^{n+1}\right. \\
&\quad \left. + (\nabla_{Q^n} \mathscr{R}^{n+2})^T \psi^{n+2}\right] \quad \text{for} \quad n = N - 2, \ldots, N_c + 1, \\
\psi^n &= -\left((\nabla_{Q^n} \mathscr{R}^n)^T\right)^{-1} \cdot \left[(\nabla_{Q^n} \mathscr{R}^{n+1})^T \psi^{n+1} + (\nabla_{Q^n} \mathscr{R}^{n+2})^T \psi^{n+2}\right] \\
&\quad \text{for} \quad n = N_c, \ldots, 1.
\end{aligned}$$

The gradient of J with respect to the design variables Y is then given by

$$\frac{\partial J}{\partial Y} = \left.\frac{\partial \mathscr{L}}{\partial Y}\right|_{\frac{\partial \mathscr{L}}{\partial Q^n} = \frac{\partial \mathscr{L}}{\partial \psi^n} = 0} = \sum_{n=N_c+1}^{N} \nabla_Y I^n(Q^n, Y) + \sum_{n=1}^{N} (\psi^n)^T \nabla_Y R(Q^n, Y).$$

In summary, the gradient is determined by the solution of the adjoint equations in reverse time and the partial derivatives of the flow residual and objective function with respect to the design variables (while Q^n is held constant). In our code the matrices $(\nabla_{Q^n} \mathscr{R}^n)^T$ are formed and stored explicitly as are the terms $(\nabla_{Q^n} I^n)^T$. The remaining terms, namely the objective function sensitivities $\nabla_Y I^n(Q^n, Y)$ and the residual sensitivities $\nabla_Y R(Q^n, Y)$, are evaluated using fourth-order centered finite differences, which are not computationally expensive.

For ESDIRK4 only Q^0 at $t = 0$ is required and the time-dependent flow solution is implicitly defined via

$$\begin{aligned}
&\mathscr{R}_k^n(Q_k^n, \ldots, Q_2^n, Q^{n-1}, Y) := \frac{Q_k^n - Q^{n-1}}{a_{kk} \Delta t_c} \\
&\quad + R(Q_k^n, Y) + \frac{1}{a_{kk}} \sum_{j=1}^{k-1} a_{kj} R(Q_j^n, Y) = 0 \\
&\quad \text{for } n = 1, \ldots, N_c \text{ and } k = 2, \ldots, 6, \\
&\mathscr{R}_k^n(Q_k^n, \ldots, Q_2^n, Q^{n-1}, Y) := \frac{Q_k^n - Q^{n-1}}{a_{kk} \Delta t} \\
&\quad + R(Q_k^n, Y) + \frac{1}{a_{kk}} \sum_{j=1}^{k-1} a_{kj} R(Q_j^n, Y) = 0 \\
&\quad \text{for } n = N_c \ + 1, \ldots, N \text{ and } k = 2, \ldots, 6,
\end{aligned}$$

with the Butcher coefficients a_{kj} given in Table 2.

The Lagrangian function is now given by

$$\begin{aligned}
\mathscr{L}(Q_2^1, \ldots, Q_6^N, \psi_2^1, \ldots, \psi_6^N, Y) = &\sum_{n=N_c+1}^{N} I^n(Q^n, Y) \\
&+ \sum_{n=1}^{N} \sum_{k=2}^{6} (\psi_k^n)^T \mathscr{R}_k^n(Q_k^n, \ldots, Q_2^n, Q^{n-1}, Y),
\end{aligned}$$

and the Lagrange multipliers ψ_k^n must now be chosen such that $\nabla_{Q_k^n}\mathscr{L} = 0$ for $n = 1,\ldots,N$ and $k = 2,\ldots,6$. Eventually this leads for $n = N,\ldots,1$ to

$$\psi_6^n = -\left(\left(\nabla_{Q_6^n}\mathscr{R}_6^n\right)^T\right)^{-1}\left[\mathscr{I}^n + \mathscr{S}^n\right],$$

$$\psi_k^n = -\left(\left(\nabla_{Q_k^n}\mathscr{R}_k^n\right)^T\right)^{-1}\left[\sum_{j=k+1}^{6}\left(\nabla_{Q_k^n}\mathscr{R}_j^n\right)^T\psi_j^n\right]$$
$$\text{for } k = 5,\ldots,2,$$

with

$$\mathscr{I}^n = \begin{cases} \left(\nabla_{Q_6^n} I^n\right)^T & \text{for } N \geqslant n \geqslant N_c, \\ 0 & \text{otherwise}, \end{cases}$$
$$\mathscr{S}^n = \begin{cases} 0 & \text{for } n = N, \\ \sum_{j=2}^{6}\left(\nabla_{Q_6^n}\mathscr{R}_j^{n+1}\right)^T\psi_j^{n+1} & \text{for } n < N. \end{cases}$$

Finally, the gradient of J with respect to the design variables Y is given by

$$\frac{\partial J}{\partial Y} = \left.\frac{\partial \mathscr{L}}{\partial Y}\right|_{\frac{\partial \mathscr{L}}{\partial Q_k^n} = \frac{\partial \mathscr{L}}{\partial \psi_k^n} = 0}$$
$$= \sum_{n=N_c+1}^{N} \nabla_Y I^n(Q^n, Y) + \sum_{n=1}^{N}\sum_{k=1}^{6}\left(\sum_{j=k}^{6}\left(\psi_j^n\right)^T\frac{a_{jk}}{a_{jj}}\right)\nabla_Y R(Q_k^n, Y)$$

with the definitions $a_{11} := 0$ and $\frac{a_{11}}{a_{11}} := 0$.

References

[1] Hayes JA, Horne WC, Soderman PT, Bent PH. Airframe noise characteristics of a 47% scale DC-10 model. AIAA Paper 97-1594; 1997.

[2] Dobrzynski W, Nagakura K, Gehlhar B, Buschbaum A. Airframe noise studies on wings with deployed high-lift devices. AIAA Paper 98-2337; 1998.

[3] Chow LC, Mau K, Remy H. Landing gears and high lift devices airframe noise research. AIAA Paper 2002-2408; 2002.

[4] Guo YP, Joshi MC. Noise characteristics of aircraft high lift systems. AIAA J 2003;51(7):1247–56.

[5] Singer BA, Guo Y. Development of computational aeroacoustics tools for airframe noise calculations. Int J Comput Fluid Dyn 2004;18(6):455–69.

[6] Hawkings DL. Noise generation by transonic open rotors. In: Mueller EA, editor. Mechanics of sound generation in flows. Berlin: Springer-Verlag; 1979.

[7] Lyrintzis AS. Surface integral methods in computational aeroacoustics – from the (CFD) near-field to the (acoustic) far-field. Aeroacoustics 2003;2(2):95–128.

[8] Singer BA, Lockard DP, Lilley GM. Hybrid acoustic predictions. Comput Math Applicat 2003;46:647–69.

[9] Farassat F, Myers MK. Extension of Kirchhoff's formula to radiation from moving surfaces. J Sound Vib 1988;123(3):451–61.

[10] Farassat F, Myers MK. The Kirchhoff formula for a supersonically moving surface. AIAA Paper 95-062; 1995.

[11] Ffowcs Williams JE, Hawkings DL. Sound generated by turbulence and surfaces in arbitrary motion. Philos Trans Royal Soc Lond 1969;A264(1151):321–42.

[12] Lighthill MJ. On sound generated aerodynamically, I: general theory. Proc Royal Soc A 1952;211:564–87.

[13] Brentner KS, Farassat F. Analytical comparison of the acoustic analogy and Kirchhoff formulation for moving surfaces. AIAA J 1998;36(8):1379–86.

[14] Wang M, Freund JB, Lele SK. Computational prediction of flow-generated sound. Annu Rev Fluid Mech 2006;38:483–512.

[15] Lockard DP. An efficient, two-dimensional implementation of the Ffowcs Williams and Hawkings equation. J Sound Vib 2000;229(4):897–911.

[16] Guo YP. Application of the Ffowcs Williams/Hawkings equation to two-dimensional problems. J Fluid Mech 2000;403:201–21.

[17] Zhang X, Rona A, Lilley G. Far-field noise radiation from an unsteady supersonic cavity flow. In: First joint CEAS/AIAA aeroacoustics conference, CEAS/AIAA 95-040; 1995.

[18] Cox JS, Brentner KS, Rumsey CL. Computation of vortex shedding and radiated sound for a circular cylinder: subcritical to transcritical Reynolds numbers. Theor Comput Fluid Dyn 1998;12:233–53.

[19] Singer BA, Brentner KS, Lockard DP, Lilley GM. Simulation of acoustic scattering from a trailing edge. J Sound Vib 2000;230(3):541–60.

[20] Singer BA, Brentner KS, Lockard DP. Computational aeroacoustic analysis of slat trailing-edge flow. AIAA J 2000;38(9):1558–64.

[21] Khorrami MR, Berkman ME, Choudhari M. Unsteady flow computations of a slat with a blunt trailing edge. AIAA J 2000;38(11):2050–8.

[22] Khorrami MR, Lockard DP. Effects of geometric details on slat noise generation and propagation. AIAA Paper 2006-2664; 2006.

[23] Choudhari MM, Khorrami MR, Lockard D, Atkins H, Lilley G. Slat cove noise modeling: a posteriori analysis of unsteady RANS simulations. AIAA Paper 2002-2468; 2002.

[24] Marsden AL, Wang M, Dennis Jr JE, Moin P. Optimal aeroacoustic shape design using the surrogate management framework. Optimization Eng 2004;5(2):235–62.

[25] Rumpfkeil MP, Zingg DW. A general framework for the optimal control of unsteady flows with applications. AIAA Paper 2007-1128; 2007.

[26] Rumpfkeil MP, Zingg DW. The optimal control of unsteady flows with a discrete adjoint method. Optimization Eng 2008. doi:10.1007/s11081-008-9035-5.

[27] Nemec M, Zingg DW. Newton–Krylov algorithm for aerodynamic design using the Navier–Stokes equations. AIAA J 2002;40(6):1146–54.

[28] Nemec M, Zingg DW, Pulliam TH. Multipoint and multi-objective aerodynamic shape optimization. AIAA J 2004;42(6):1057–65.

[29] Nadarajah S, Jameson A. Optimum shape design for unsteady flows with time-accurate continuous and discrete adjoint methods. AIAA J 2007;45(7):1478–91.

[30] Mani K, Mavriplis DJ. Unsteady discrete adjoint formulation for two-dimensional flow problems with deforming meshes. AIAA J 2008;46(6):1351–64.

[31] Yamaleev NK, Diskin B, Nielsen EJ. Adjoint-based methodology for time-dependent optimization. AIAA Paper 2008-5857; 2008.

[32] Choi S, Potsdam M, Lee K, Iaccarino G, Alonso JJ. Helicopter rotor design using a time-spectral and adjoint-based method. AIAA Paper 2008-5810; 2008.

[33] Mavriplis DJ. Solution of the unsteady discrete adjoint for three-dimensional problems on dynamically deforming unstructured meshes. AIAA Paper 2008-727; 2008.

[34] Pueyo A, Zingg DW. Efficient Newton–Krylov solver for aerodynamic computations. AIAA J 1998;36(11):1991–7.

[35] Isono S, Zingg DW. A Runge–Kutta–Newton–Krylov algorithm for fourth-order implicit time marching applied to unsteady flows. AIAA Paper 2004-0433; 2004.

[36] Tabesh M, Zingg DW. Higher-order implicit time-marching methods using a Newton–Krylov algorithm. AIAA Paper 2009-0164; 2009.

[37] Pulliam TH. Efficient solution methods for the Navier–Stokes equations. Lecture notes for the Von Karman Institute for fluid dynamics lecture series; 1986.

[38] Spalart PR, Allmaras SR. A one-equation turbulence model for aerodynamic flows. AIAA Paper 92-0439; 1992.

[39] Bijl H, Carpenter MH, Vatsa VN. Time integration schemes for the unsteady Navier–Stokes equations. AIAA Paper 2001-2612; 2001.

[40] Saad Y, Schultz MH. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J Sci Stat Comput 1986;7(3):856–69.

[41] Meijerink JA, van der Vorst HA. An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. Math Comput 1977;31(137):148–62.

[42] Greschner B, Yu C, Zheng S, Zhuang M, Wang ZJ, Thiele F. Knowledge based airfoil aerodynamic and aeroacoustic design. AIAA Paper 2005-2968; 2005.

[43] Morfey CL, Wright MCM. Extensions of Lighthills acoustic analogy with application to computational aeroacoustics. Proc Royal Soc A 2007;463:2101–27.

[44] van der Vorst H. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. SIAM J Sci Stat Comput 1992;13:631–44.

[45] Hinze M, Walther A, Sternberg J. Discrete approximation schemes for reduced gradients and reduced Hessians in Navier–Stokes control utilizing an optimal memory-reduced procedure for calculating adjoints. Opt Control Applicat Methods 2006;27(1):19–40.

[46] Byrd RH, Lu P, Nocedal J, Zhu C. A limited memory algorithm for bound constrained optimization. SIAM J Sci Comput 1995;5(16):1190–208.

[47] Zhu C, Byrd RH, Lu P, Nocedal J. L-BFGS-B: a limited memory FORTRAN code for solving bound constrained optimization problems. Tech. rep. NAM-11, EECS Department, Northwestern University; 1994.

[48] Nadarajah S, McMullen M, Jameson A. Optimum shape design for unsteady flows using time accurate and non-linear frequency domain methods. AIAA Paper 2003-3875; 2003.

[49] Wang L, Mavriplis DJ, Anderson WK. Unsteady discrete adjoint formulation for high-order discontinuous Galerkin discretizations in time-dependent flow problems. AIAA Paper 2010-367; 2010.

Please cite this article in press as: Rumpfkeil MP, Zingg DW. A hybrid algorithm for far-field noise minimization. Comput Fluids (2010), doi:10.1016/j.compfluid.2010.05.006