ECCOMAS Computational Fluid Dynamics Conference 2001
Swansea, Wales, UK, 4-7 September 2001

ADJOINT METHODS FOR AERONAUTICAL DESIGN

Michael B. Giles*

*Oxford University Computing Laboratory,
Wolfson Building, Parks Road, OX1 3QD, UK.
E-mail: giles@comlab.ox.ac.uk

Key words: Compressible flow, adjoint equations, design optimisation, aeroelasticity, structural vibration.

Abstract. *This paper gives an overview of the use of adjoint equations in aeronautical design optimisation to obtain the sensitivity of an objective function to changes in any number of design variables. Both the continuous and the discrete adjoint approach are outlined and the author's preference for the latter is explained.*

1 Introduction

There is a long history of the use of adjoint equations in optimal control theory [27]. In fluid dynamics, the first use of adjoint equations for design was by Pironneau [32]. However, within the field of aeronautical computational fluid dynamics, it is Jameson who has applied the methods of optimal control theory to formulate optimal design methods. The term 'optimal' refers to the fact that one is trying to find the geometry which minimises some objective function, such as the drag. In a sequence of papers [20, 21, 23] Jameson has developed the adjoint approach for potential flow, the Euler equations and the Navier-Stokes equations. The complexity of the applications within these papers also progressed from 2D airfoil optimisation, to 3D wing design and finally to complete aircraft configurations [22, 33, 34]. A number of other research groups have also developed adjoint CFD codes [26, 36, 4, 3, 5] using the same 'continuous' approach in which the first step is to linearise the original partial differential equations. The adjoint p.d.e. and appropriate boundary conditions are then formulated, and finally the equations are discretised.

The alternative 'discrete' approach takes a discretisation of the Euler or Navier-Stokes equations, linearises the discrete equations and then uses the transpose of the linear operator to form the adjoint problem. This approach has been developed by Elliott [7], Anderson [31, 1], Mohammadi [28, 29] and Kim [25], and it is the approach favoured by the present author.

This paper outlines both approaches, emphasising the underlying similarity in their mathematics. The adjoint theory is presented firstly in the context of linear algebra, in which it is most easily understood. This is the basis for the discrete adjoint CFD approach in which one works with the algebraic equations that come from the discretisation of the original fluid dynamic equations. The paper then treats the extension to p.d.e.'s as used in Jameson's continuous adjoint approach. Here the emphasis is on the construction of the adjoint p.d.e. and its boundary conditions, including the manner in which geometric perturbations are introduced.

The pros and cons of the two approaches are then discussed, but in the end it is a matter of personal judgement. There are advocates for each approach, but no suggestions that one approach is clearly better than the other. The paper concludes with a few simple numerical test cases illustrating the computation of lift and mass flow sensitivities.

For further information, see the excellent review by Newman *et al* [30] which surveys both continuous and adjoint methods, and the papers by Giles [12, 11], and Giles & Pierce [17] which present a more extensive introduction to the adjoint approach to design and some of the related design optimisation issues.

2 Discrete adjoint approach

2.1 Fundamental linear algebra

Suppose one wishes to evaluate the vector dot product $g^T u$, with u being the solution of the linear system of equations

$$Au = f,$$

for some given matrix A and vector f. An equivalent dual form is to evaluate $v^T f$ where the adjoint solution v satisfies the linear system of equations

$$A^T v = g.$$

Note the use of the transposed matrix A^T, and the interchange in the roles of f and g. The equivalence of the two forms is easily proved as follows,

$$v^T f = v^T A u = (A^T v)^T u = g^T u.$$

Given a single f and a single g, nothing would be gained (or lost) by using the dual form. However, if we want the value of the objective function for p different values of f, and m different value of g, the standard approach needs the solution of p different primal equations, whereas the adjoint approach needs the solution of m different adjoint calculations. Therefore, the adjoint approach is much cheaper when $m \ll p$.

2.2 Design sensitivities

Given a set of design variables, α, which control the geometry of the airfoil, wing or aircraft being designed, and a set of flow variables at discrete grid points, U, the aim is to determine the sensitivity of a single objective function $J(U, \alpha)$ to changes in α. The discrete flow equations, together with the boundary conditions, can be expressed as

$$R(U, X(\alpha)) = 0,$$

where X is the vector of grid point coordinates which depends on α. For a single design variable, we can linearise about a baseline geometry and flow solution to get

$$\frac{dJ}{d\alpha} = \frac{\partial J}{\partial U}\, \frac{dU}{d\alpha} + \frac{\partial J}{\partial \alpha}.$$

The flow sensitivity $dU/d\alpha$ satisfies the linearised flow equations

$$\frac{\partial R}{\partial U}\, \frac{dU}{d\alpha} + \frac{\partial R}{\partial \alpha} = 0.$$

By defining

$$u = \frac{dU}{d\alpha}, \quad A = \frac{\partial N}{\partial U}, \quad g^T = \frac{\partial J}{\partial U}, \quad f = -\frac{\partial N}{\partial \alpha}$$

we can convert this into the standard form

$$\frac{dJ}{d\alpha} = g^T u + \frac{\partial J}{\partial \alpha},$$

subject to

$$Au = f.$$

The term $\partial J/\partial\alpha$ is relatively easy to evaluate. The term $g^T u \equiv v^T f$ can be computed either by the direct approach, solving $Au = f$, or by the adjoint approach, solving $A^T v = g$. For a single design variable there would be no benefit in using the adjoint approach, but for multiple design variables, each has a *different* f, but the *same* g, so the adjoint approach is computationally much more efficient.

2.3 Implementation issues

The above description of the discrete adjoint approach makes it seem straightforward, and this is one of the strengths of the discrete approach. However, in implementing it, a number of important issues arise, of which the most important are:

- Programming of adjoint matrix-vector product

  We have written by hand our adjoint code [13] to evaluate $A^T v$, but this is not a very easy task. A better alternative may be to follow Mohammadi [28, 29] in using automatic differentiation software [8, 9, 19], however this too is not without its difficulties. For the evaluation of $f = -\partial N/\partial\alpha$ we have used the "complex variable technique" [35] used by Anderson *et al* [2]. This is a very effective technique which is easily implemented.

- Solid wall boundary conditions for node-based discretisations

  The implementation of solid wall boundary conditions for node-based discretisations involves the discarding of momentum residuals at wall nodes, to be replaced by a no-flux or no-slip condition for inviscid and viscous cases, respectively. In addition, the discarded momentum residuals sometimes form part of the functional to be evaluated. Both of these features introduce some additional complexity in formulating the adjoint problem [13].

- Iterative solution of adjoint equations

  The eigenvalues of A^T are exactly the same as those of A. Therefore, many standard iterative methods, such as the GMRES method used by Anderson [3], are guaranteed to converge with the same asymptotic rate of convergence as for the original nonlinear code. In our work [13], we use a special form of preconditioned time-marching with multigrid, and obtain exactly the same convergence history for the sensitivity from the adjoint code as we do from a linearised flow perturbation code. This also provides a very useful check on the correctness of our programming.

3 Continuous adjoint approach

3.1 Fundamental theory

Duality in the case of p.d.e.'s is a natural extension of duality in the linear algebra formulation. Using (V,U) to denote an integral inner product over some domain Ω,

$$(V,U) \equiv \int_\Omega V^T U \; dx,$$

suppose one wants to evaluate the functional (g,u), where u is the solution of the p.d.e.

$$Lu = f,$$

on the domain Ω subject to homogeneous boundary conditions on the boundary $\partial\Omega$.

Using the adjoint formulation, the identical functional takes the form (v,f) where v is the solution of the adjoint p.d.e.

$$L^* v = g,$$

plus appropriate homogeneous adjoint b.c.'s. The adjoint operator L^* is defined by the identity

$$(V, LU) = (L^* V, U),$$

which must hold for all functions V,U satisfying the respective homogeneous boundary conditions. Given the definitions, the proof of the equivalence of the two forms of the problem is trivial

$$(v,f) = (v, Lu) = (L^* v, u) = (g,u).$$

Thus far, the theory looks extremely similar to the linear algebra behind the discrete approach. However, in general, the objective function of interest involves integrals over the boundary, rather than over the domain, and the boundary conditions are not homogeneous. To handle this, the following more general form of the adjoint identity is required.

$$(V, LU)_\Omega + (C^* V, BU)_{\partial\Omega} = (L^* V, U)_\Omega + (B^* V, CU)_{\partial\Omega}$$

for all functions U,V, with the notation $(.,.)_{\partial\Omega}$ denoting an inner product over the boundary. B and C are both boundary operators (possibly involving normal derivatives) given in the definition of the original problem. B^* and C^* are the corresponding adjoint boundary operators which can be found by integration by parts. Using this adjoint identity, it follows immediately that

$$(v,f)_\Omega + (C^* v, f_2)_{\partial\Omega} = (g,u)_\Omega + (g_2, Cu)_{\partial\Omega}$$

when the primal problem is

$$Lu = f \text{ in } \Omega, \quad \text{and} \quad Bu = f_2 \text{ on } \partial\Omega,$$

and the adjoint problem is

$$L^* v = g \text{ in } \Omega, \quad \text{and} \quad B^* v = g_2 \text{ on } \partial\Omega.$$

There are some restrictions on what can be imposed as b.c.'s and objective functions. The analysis is complicated (see [23] and [14] for details) but it reveals that on a solid surface, the boundary integral term in the objective function must be a weighted integral of the linear perturbation in the pressure when using the Euler equations. Similarly, for the Navier-Stokes equations it must be a weighted integral of the linear perturbation in the normal and tangential forces on the surface, and either the heat flux or the surface temperature (depending whether one is specifying the surface temperature or adiabatic conditions, respectively).

3.2 Design sensitivities

The most complicated step in the continuous approach to design sensitivities is formulating the linearised flow equations. In two dimensions, Jameson uses curvilinear coordinates (ξ, η) corresponding to grid lines of a structured grid, with the airfoil surface being defined as $\eta\!=\!0$ [21]. The transformed Euler equations can be written as

$$\frac{\partial}{\partial \xi}\left(F\frac{\partial y}{\partial \eta} - G\frac{\partial x}{\partial \eta}\right) + \frac{\partial}{\partial \eta}\left(-F\frac{\partial y}{\partial \xi} + G\frac{\partial x}{\partial \xi}\right) = 0,$$

where F and G represent the usual inviscid fluxes in the x and y directions. A small perturbation $\tilde{\alpha}$ to a design parameter produces changes such as

$$\begin{aligned} F \quad &\longrightarrow \quad F + \frac{\partial F}{\partial U}\,\frac{dU}{d\alpha}\,\tilde{\alpha} \\ \frac{\partial x}{\partial \eta} \quad &\longrightarrow \quad \frac{\partial x}{\partial \eta} + \frac{\partial^2 x}{\partial \eta \partial \alpha}\,\tilde{\alpha}. \end{aligned}$$

Terms *not* depending on $\tilde{\alpha}$ all cancel, and terms depending on $\tilde{\alpha}^2$ are neglected. Hence, we get the linearised equations,

$$\begin{aligned} \frac{\partial}{\partial \xi}\left(\left(A\frac{\partial y}{\partial \eta} - B\frac{\partial x}{\partial \eta}\right)u\right) + \frac{\partial}{\partial \eta}\left(\left(-A\frac{\partial y}{\partial \xi} + B\frac{\partial x}{\partial \xi}\right)u\right) \quad &= \\ -\frac{\partial}{\partial \xi}\left(F\frac{\partial^2 y}{\partial \eta \partial \alpha} - G\frac{\partial^2 x}{\partial \eta \partial \alpha}\right) - \frac{\partial}{\partial \eta}\left(-F\frac{\partial^2 y}{\partial \xi \partial \alpha} + G\frac{\partial^2 x}{\partial \xi \partial \alpha}\right)&, \end{aligned}$$

where

$$A = \frac{\partial F}{\partial U}, \quad B = \frac{\partial G}{\partial U}, \quad u = \frac{dU}{d\alpha}.$$

The boundary condition on an inviscid wall is that there is no flow normal to the surface $\eta\!=\!0$. This remains true as α changes but one needs to consider the linearised perturbation to the unit normal, which eventually leads an inhomogeneous boundary term.

For complex geometries, it is often not possible to use structured grids. However, the same idea of using perturbed curvilinear coordinates can be extended to unstructured grids [11, 17].

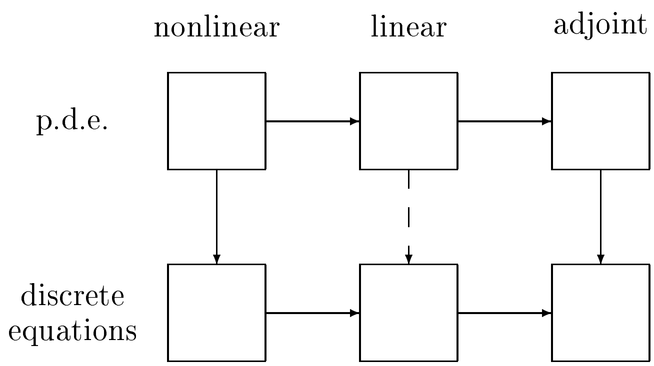

Figure 1: Alternative approaches to forming discrete adjoint equations

3.3 Implementation issues

With the continuous adjoint approach, the discretisation of the adjoint equations can be carried out without regard for the discretisation of the nonlinear flow problem. However, the standard issues of accuracy, stability and convergence remain to be addressed.

When considering flows with shocks, the analytic formulation should treat the shocks as discontinuities across which the Rankine-Hugoniot shock jump relations are enforced. This leads to the important result that the adjoint variables are continuous across the shock and that an additional adjoint boundary condition must be imposed [16]. However, imposing such a b.c. would be complicated, as it would require the automatic identification of the shock location in the nonlinear flow calculation, so in practice, the standard practice is not to enforce this condition. Quasi-one-dimensional results have demonstrated that the inclusion of numerical smoothing automatically leads to satisfaction of the adjoint boundary condition at the shock [15], and results in higher dimensions do not indicate any particular anomalies.

4 Relative advantages of two approaches

The difference between the discrete and continuous approaches is shown schematically in Figure 1. In both cases one obtains a set of discrete adjoint equations. In the discrete approach one starts by discretising the nonlinear p.d.e.; these equations are then linearised and transposed. In the the continuous adjoint approach, the discretisation is the final step, after first linearising and forming the adjoint problem. One could even follow an intermediate path, linearising the original equations, discretising them and then taking the transpose. In principle, if each of the steps is performed correctly, and all of the solutions are sufficiently smooth (*e.g.* no shocks) then in the limit of infinite grid resolution all three approaches should be consistent and converge to the correct analytic value for design sensitivity.

However, there are important conceptual differences between the different approaches, and for finite resolution grids there will be differences in the computed results. In the author's opinion, the main advantages of the discrete approach are:

- Creation of the adjoint program is conceptually straightforward, and in the future will hopefully be relatively easy using automatic differentiation software.
- Iterative methods based on those used for the solution of the nonlinear flow equations are guaranteed to be stable, and with the same highly-optimised rate of convergence.
- There are numerous self-consistency checks which can be performed, comparing nonlinear flux routines with their adjoint counterparts, to identify programming errors.

On the other hand, the advantages of the continuous approach are:

- The physical significance of the adjoint variables and the role of adjoint b.c.'s is much clearer. Only by constructing the adjoint flow equations can one develop a good understanding of the nature of adjoint solutions, such as the continuity at shocks, the logarithmic singularity at a sonic point in quasi-1D flows [16] but not in 2D or 3D (in general) and the inverse square-root singularity along the stagnation streamline upstream of an airfoil in 2D [14].
- The adjoint program is simpler and requires less memory because one is free to discretise the adjoint p.d.e. in any consistent way.

It remains an open question as to whether one approach is better when there are nonlinear discontinuities such as shocks. For quasi-1D Euler calculations, both approaches give numerical results which converge to the analytic solution [16]. For the discrete approach, this follows because the integrated pressure can be proved to be predicted with second order accuracy [10]. The linearised discretisation should therefore yield perturbations to the integral of pressure that are at least first-order accurate. The discrete adjoint formulation, which is constructed using this linearised operator, must therefore behave correctly to first order at the shock. For the continuous approach, in the absence of explicit enforcement of the correct adjoint b.c. at the shock, the correct asymptotic behaviour can be explained as the effect of numerical smoothing, given that the correct analytic solution is the only smooth solution at the shock [15].

However, in 2D and 3D there is no proof of second order accuracy for quantities such as lift and drag, and there is a discontinuity in the gradient of the adjoint variables at the location of the shock. Therefore it remains an open question as to whether either approach will give a consistent approximation to the gradient of the objective function in the limit of infinite grid resolution. Numerical results for test cases with strong shocks indicate there may be a problem with the discrete approach [13], but results using both approaches suggest that any inconsistency is very small when the shocks are weak.

5 Nonlinear optimisation

Returning to the design problem, the aim is to find the set of design variables α which minimise the nonlinear objective function $J(U,\alpha)$, where U is an implicit function of α through the flow equations

$$N(U,\alpha) = 0.$$

These nonlinear flow equations and the corresponding linear adjoint equations are both large systems which are usually solved by an iterative procedure.

There are two main optimisation strategies using the design sensitivities obtained from the adjoint problem. The first is to use a simple steepest descent algorithm,

$$\Delta\alpha = -\epsilon \, \frac{dJ}{d\alpha},$$

where ϵ controls the step size. The advantage of this method is that partially-converged flow and adjoint solutions may be used to evaluate the gradients as long as these gradients are properly preconditioned (through numerical smoothing) prior to updating α [21]. As a result, the cost per design cycle is relatively low.

In the second approach, approximations to the Hessian matrix of second derivatives

$$\frac{d^2 J}{d\alpha_i d\alpha_j},$$

are used to speed convergence via a quasi-Newton procedure such as BFGS [18]. This method requires more accurate flow and adjoint solutions, which must generally be converged almost fully during each design iteration. As a result, the cost of each design cycle is significantly increased.

The relative efficiency and robustness of the two strategies is still subject to debate, but the recent paper by Jameson and Vassberg [24] comparing the two techniques presents convincing support for the first approach.

6 Numerical results

Figures 2–4 give some examples of adjoint results from another paper [13]. Figure 2 is an inviscid test case. The symbols show the variation in lift coefficient with angle of attack for a NACA0012 airfoil at a freestream Mach number of 0.68. Each of the lines has a slope given by the lift sensitivity calculated by the adjoint code based on the nonlinear flow conditions at the angle of attack at the central point.

Figure 3 is similar, but for a viscous test case, the RAE2822 airfoil at a freestream Mach number of 0.725 and a Reynolds number of 6.5×10^6. The Spalart-Allmaras turbulence model is used, and the adjoint code incorporates the linearisation of the turbulence model. Again there is good agreement between the nonlinear and adjoint results.

Finally, Figure 5 shows an example of a different kind of adjoint calculation. This is a test case of unsteady flow over a cascade of flat plate airfoils, with the unsteadiness

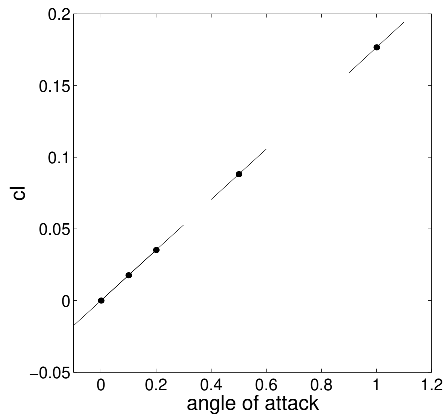

Figure 2: C_l vs. angle of attack for a NACA 0012 profile at $M = 0.68$.

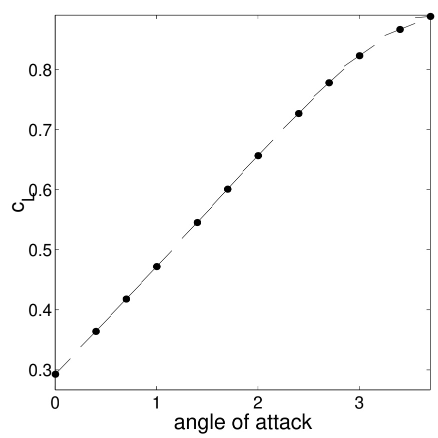

Figure 3: Lift vs. angle of attack for a RAE 2822 profile at $M = 0.725$, $\mathrm{Re} = 6.5 \times 10^6$.

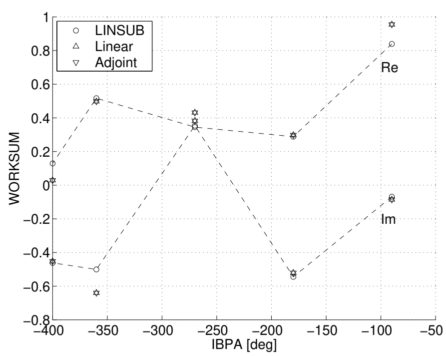

Figure 4: Bending mode worksum components due to wake interaction, versus interblade phase angle associated with wake pitch.

being caused by incoming wakes with a sinusoidal profile. This test case is relevant to the problems of structural vibration due to forced response and flutter in turbomachinery. The standard analysis uses harmonic linear unsteady flow analysis to compute the unsteady flow for a single frequency of unsteady forcing. This is expressed as the real part of a complex amplitude multiplying a harmonic unsteady term,

$$u(x,y,t) = \mathcal{R}\left\{\hat{u}(x,y)\exp(i\omega t)\right\}.$$

The key output is a complex quantity called the "worksum" which represents the generalised force for a particular vibration mode within the context of classical Lagrangian mechanics. The figure shows the real and imaginary components of this quantity for the bending mode of vibration, and its variation as a function of the interblade phase angle which is related to the pitch of the incoming wakes. The adjoint calculation computes the same quantity using the complex conjugate transpose of the linear harmonic discrete matrix [13, 6]. The linear and adjoint codes produces identical values for the worksum, and they agree well with the values produced by another code LINSUB based on semi-analytic theory [37]. The benefit of the adjoint approach for such unsteady problems is that it can give the level of forced response for any incoming wake of a particular frequency, which is useful in certain design applications aiming to minimise forced response vibrations [6]. There is also the potential of using it for the design of blades which a reduced susceptibility to flutter.

7 Conclusions

This paper has reviewed the underlying theory for optimal design using adjoint methods to obtain the sensitivity of an objective function with respect to a large number of design variables. Both the continuous and the discrete approaches have been covered, and their relative strengths have been commented on. It is hoped that this will encourage and help others to develop adjoint techniques as an integral part of engineering design systems. Although the ideas have been presented in the context of aeronautical design, the ideas are equally relevant to any area of engineering design involving large numbers of continuous design variables.

Acknowledgements

This research has been supported by the Engineering and Physical Sciences Research Council under grant GR/L95700, and by Rolls-Royce plc (technical monitor: Leigh Lapworth) DERA (technical monitor: John Calvert), and BAESystems plc (technical monitor: David Standingford). I would also like to acknowledge the contributions of M.S. Campobasso, M.C. Duta, J.-D. Müller and N.A. Pierce to the development and validation of the adjoint HYDRA codes.

REFERENCES

[1] W.K. Anderson and D.L. Bonhaus. Airfoil design on unstructured grids for turbulent flows. *AIAA J.*, 37(2):185–191, 1999.

[2] W.K. Anderson, J. Newman, D. Whitfield, and E. Nielsen. Sensitivity analysis for the Navier-Stokes equations on unstructured grids using complex variables. AIAA Paper 99-3294, 1999.

[3] W.K. Anderson and V. Venkatakrishnan. Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation. *Comput. & Fluids*, 28(4-5):443–480, 1999.

[4] O. Baysal and M. Eleshaky. Aerodynamic design optimization using sensitivity analysis and computational fluid dynamics. *AIAA J.*, 30(3):718–725, 1992.

[5] A. Dadone and B. Grossman. Progressive optimization of inverse fluid dynamic design problems. *Comput. & Fluids*, 29(1), 2000.

[6] M. Duta, M.B. Giles, and M.S. Campobasso. The harmonic adjoint approach to unsteady turbomachinery design. ICFD Conference, Oxford, 2001.

[7] J. Elliott and J. Peraire. Practical 3D aerodynamic design and optimization using unstructured meshes. *AIAA J.*, 35(9):1479–1485, 1997.

[8] C. Faure and Y. Papegay. Odyssée User's Guide version 1.7. Technical Report RT-0224, INRIA, Sophia-Antipolis, 1998. See `www.inria.fr/safir/SAM/Odyssee/odyssee.html`.

[9] R. Giering and T. Kaminski. Recipes for adjoint code construction. *ACM Trans. Math. Software*, 24(4):437–474, 1998.

[10] M.B. Giles. Analysis of the accuracy of shock-capturing in the steady quasi-1D Euler equations. *Comput. Fluid Dynamics J.*, 5(2):247–258, 1996.

[11] M.B. Giles. Aerodynamic design optimisation for complex geometries using unstructured grids. Lecture notes for VKI Lecture Course on Inverse Design. Technical Report NA97/08, Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD, 1997.

[12] M.B. Giles. Aerospace design: a complex task. Lecture notes for VKI Lecture Course on Inverse Design. Technical Report NA97/07, Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD, 1997.

[13] M.B. Giles, M.C. Duta, and J.-D. Müller. Adjoint code developments using the exact discrete approach. AIAA Paper 2001-2596, 2001.

[14] M.B. Giles and N.A. Pierce. Adjoint equations in CFD: duality, boundary conditions and solution behaviour. AIAA Paper 97-1850, 1997.

[15] M.B. Giles and N.A. Pierce. On the properties of solutions of the adjoint Euler equations. In M. Baines, editor, *Numerical Methods for Fluid Dynamics VI*. ICFD, Jun 1998.

[16] M.B. Giles and N.A. Pierce. Analytic adjoint solutions for the quasi-one-dimensional Euler equations. *J. Fluid Mech.*, 426:327–345, 2001.

[17] M.B. Giles and N.A. Pierce. An introduction to the adjoint approach to design. *Flow, Turbulence and Control*, to appear, 2001.

[18] P. Gill, W. Murray, and M. Wright. *Practical optimization*. Academic Press, 1981.

[19] A. Griewank. *Evaluating derivatives : principles and techniques of algorithmic differentiation*. SIAM, 2000.

[20] A. Jameson. Aerodynamic design via control theory. *J. Sci. Comput.*, 3:233–260, 1988.

[21] A. Jameson. Optimum aerodynamic design using control theory. In M. Hafez and K. Oshima, editors, *Computational Fluid Dynamics Review 1995*, pages 495–528. John Wiley & Sons, 1995.

[22] A. Jameson. Re-engineering the design process through computation. *J. Aircraft*, 36:36–50, 1999.

[23] A. Jameson, N. Pierce, and L. Martinelli. Optimum aerodynamic design using the Navier-Stokes equations. *J. Theor. Comp. Fluid Mech.*, 10:213–237, 1998.

[24] A. Jameson and J. Vassberg. Studies of alternate numerical optimization methods applied to the brachistochrone problem. OptiCON '99 Conference, 1999.

[25] H.-J. Kim, D. Sasaki, S. Obayashi, and K. Nakahashi. Aerodynamic optimization of supersonic transport wing using unstructured adjoint method. Proceedings of the ICCFD conference, Kyoto, 2000.

[26] V.M. Korivi, A.C. Taylor III, and G.W. Hou. Sensitivity analysis, approximate analysis and design optimization for internal and external viscous flows. AIAA Paper 91-3083, 1991.

[27] J.L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer-Verlag, 1971. Translated by S.K Mitter.

[28] B. Mohammadi. Practical applications to fluid flows of automatic differentiation for design problems. VKI Lecture Series 1997-05 on Inverse Design, 1997.

[29] B. Mohammadi and O. Pironneau. Mesh adaption and automatic differentiation in a CAD-free framework for optimal shape design. *Internat. J. Numer. Methods Fluids*, 30(2):127–136, 1999.

[30] J.C. Newman, A.C. Taylor, R.W. Barnwell, P.A. Newman, and G. J.-W. Hou. Overview of sensitivity analysis and shape optimization for complex aerodynamic configurations. *J. Aircraft*, 36(1):87–96, 1999.

[31] E. Nielsen and W.K. Anderson. Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations. *AIAA J.*, 37(11):957–964, 1999.

[32] O. Pironneau. On optimum design in fluid mechanics. *J. Fluid Mech.*, 64:97–110, 1974.

[33] J. Reuther, A. Jameson, J.J. Alonso, M.J. Remlinger, and D. Saunders. Constrained multipoint aerodynamic shape optimisation using an adjoint formulation and parallel computers, part 1. *J. Aircraft*, 36(1):51–60, 1999.

[34] J. Reuther, A. Jameson, J.J. Alonso, M.J. Remlinger, and D. Saunders. Constrained multipoint aerodynamic shape optimisation using an adjoint formulation and parallel computers, part 2. *J. Aircraft*, 36(1):61–74, 1999.

[35] W. Squire and G. Trapp. Using complex variables to estimate derivatives of real functions. *SIAM Rev.*, 10(1):110–112, 1998.

[36] S. Ta'asan, G. Kuruvila, and M.D. Salas. Aerodynamic design and optimization in one shot. AIAA Paper 92-0025, 1992.

[37] D. S. Whitehead. Classic two-dimensional methods. In M. Platzer and F. O. Carta, editors, *Aeroelasticity in Axial-Flow Turbomachines, AG-298*, volume 1. AGARD, 1987.