

Comparison of two adjoint equation approaches with respect to boundary-condition treatments for the quasi-1D Euler equations

G.F. Duivesteijn¹, H. Bijl, B. Koren, E.H. van Brummelen

*Faculty of Aerospace Engineering
Delft University of Technology
P.O. Box 5058, 2600 GB Delft, The Netherlands*
¹ email: g.f.duivesteijn@lr.tudelft.nl

SUMMARY

This work compares a discrete and analytical adjoint equation method with respect to boundary-condition treatments applied to the quasi-1D Euler equations. Flux evaluation of the primal problem is done by either a Godunov-type scheme or a central scheme. For our future goal, solving fluid-structure problems, the discrete approach seems preferable. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: adjoint equation, boundary conditions, quasi-1D Euler equation

1. INTRODUCTION

For efficient computation of large-scale fluid-flow problems, an efficient error estimation and grid adaptation algorithm is desirable. Traditional error estimation or grid adaptation may not suffice, since they are insufficiently related to relevant engineering quantities. The dual formulation can be used as an efficient a-posteriori error estimation in the quantity of interest. However, derivation of the dual problem, especially the accompanying boundary conditions, is not a trivial task.

Two ways of formulating the dual problem exist: analytical [1, 2] and discrete [5, 6]. This paper gives an outline of the boundary-condition derivation for both methods. For the analytical method, carefully crafted boundary conditions are needed. For the numerical method, imposing strong or weak boundary conditions to the primal problem has a great influence on the implicitly given boundary conditions for the numerical dual problem. Also, the dependence of the dual solution on the chosen flux evaluator in the primal solution will be discussed.

Several techniques exist to evaluate the Jacobian needed for setting up the dual problem. For the analytical dual formulation, differentiation is done by hand or by means of a symbolic mathematical software program. For the discrete dual problem, differentiation can be done by divided derivatives or with automatic differentiation. The different methods will be briefly discussed.

2. FLOW EQUATIONS AND OUTPUT FUNCTIONAL

To illustrate different ways to implement and use the adjoint method, an inviscid, compressible gas flow through a convergent-divergent channel will be modelled by the quasi-1D Euler equations:

$$\int_{\Omega} \frac{\partial Aq}{\partial t} d\Omega + \oint_{\partial\Omega} AF(q) d\partial\Omega - \int_{\Omega} \frac{dA}{dx} J(q) d\Omega = 0, \quad (1)$$

with

$$q = \begin{pmatrix} \rho \\ \rho u \\ \rho E \end{pmatrix}, \quad F(q) = \begin{pmatrix} \rho u \\ (\rho u^2 + p) \\ \rho u \left(E + \frac{p}{\rho} \right) \end{pmatrix}, \quad J(q) = \begin{pmatrix} 0 \\ p \\ 0 \end{pmatrix}. \quad (2)$$

with A the height of the channel [2, 5]. The output functional considered is

$$I = \int_{-1}^1 p(q) d\Omega, \quad (3)$$

where $x = -1$ and $x = 1$ are the coordinates of the inlet and outlet, respectively.

For solving the primal problem, a structured-grid, cell-centred finite-volume solver has been developed. We consider Lax, linearised Godunov, Osher or Jameson flux evaluators at the cell faces. Depending on the chosen flux evaluator, finding the steady state solution of the non-linear system of equations is done either directly for the steady equations by a global Newton iteration method, or indirectly, via time marching the unsteady equations to steady state with forward Euler or four-stage Runge-Kutta.

3. CONTINUOUS ADJOINT APPROACH

With the method, described in [2], the analytical adjoint equations will be discretised on the same grid as the primal flow equations. First, the quasi-1D Euler equations have to be linearised:

$$Lq' = \left(\frac{\partial AF(q)}{\partial q} q' \right)_x - \frac{dA}{dx} \frac{\partial J(q)}{\partial q} q' = r'. \quad (4)$$

The change in the output functional due to small perturbations in the flow solution can be written as

$$I' = \int_{\Omega} \frac{\partial p}{\partial q} q' dx. \quad (5)$$

The influence of the change in solution on the functional can be determined by the adjoint equation, which can be analytically derived by partial integration:

$$\begin{aligned} & \int_{-1}^1 v \left(\left(\frac{\partial AF(q)}{\partial q} q' \right)_x - \frac{dA}{dx} \frac{\partial J(q)}{\partial q} q' \right) dx = \\ & \left[v \frac{\partial AF(q)}{\partial q} q' \right]_{-1}^1 - \int_{-1}^1 v_x \frac{\partial AF(q)}{\partial q} q' + v \frac{dA}{dx} \frac{\partial J(q)}{\partial q} q' dx = \\ & \left[v \frac{\partial AF(q)}{\partial q} q' \right]_{-1}^1 - \int_{-1}^1 q' \left(\left[\frac{\partial AF(q)}{\partial q} \right]^T v_x + \left[\frac{dA}{dx} \frac{\partial J(q)}{\partial q} \right]^T v \right) dx = \int_{-1}^1 v r' dx. \end{aligned} \quad (6)$$

From the derivation of the adjoint equations, shown in (6), it follows that the adjoint variables may be defined in such a way that

$$q'^T \left[\frac{\partial AF(q)}{\partial q} \right]^T v = 0, \quad (7)$$

so that the dependence of q has been removed. Then, the adjoint equation corresponding to (4) is

$$L^* v \equiv - \left[\frac{\partial AF(q)}{\partial q} \right]^T v_x - \left[\frac{dA}{dx} \frac{\partial J(q)}{\partial q} \right]^T v = \frac{\partial p}{\partial q}. \quad (8)$$

Equation (8) can be written by using Jacobians based on the non-conservative flow variables $q = (\rho, u, p)^T$, so that the adjoint equation becomes

$$A \begin{pmatrix} q_2 & q_2^2 & \frac{1}{2} q_2^3 \\ q_1 & 2q_1 q_2 & \frac{\gamma}{\gamma-1} q_3 + \frac{3}{2} q_1 q_2^2 \\ 0 & 1 & \frac{\gamma}{\gamma-1} q_2 \end{pmatrix} \frac{dv}{dx} = - \begin{pmatrix} 0 \\ 0 \\ 1 + \frac{dA}{dx} v_2 \end{pmatrix}. \quad (9)$$

For this system of adjoint o.d.e's, complementary boundary conditions have to be defined. As an illustration, the derivation of boundary conditions for the subsonic case will be given. Analogous to [2], suppose $H = E + \frac{p}{\rho} = \frac{\gamma}{\gamma-1} \frac{p}{\rho} + \frac{1}{2} u^2 = H_{\text{in}}$, $p_t = p_{t_{\text{in}}}$ at the inflow and $p = p_{\text{out}}$ at the outflow. Hence, rewritten in terms of q , one gets

$$\left. \begin{aligned} \frac{\gamma}{\gamma-1} \frac{q_3}{q_1} + \frac{1}{2} q_2^2 &= H_{\text{in}} \\ q_3 + \frac{1}{2} q_1 q_2^2 &= p_{t_{\text{in}}} \end{aligned} \right\}, \quad x = -1, \quad (10)$$

$$q_3 = p_{\text{out}}, \quad x = 1. \quad (11)$$

At the boundaries, perturbations in the prescribed states are not permitted:

$$\left. \begin{aligned} \frac{\gamma}{\gamma-1} \left(\frac{q'_3}{q_1} - \frac{q_3}{q_1^2} q'_1 \right) + q_2 q'_2 &= 0 \\ q'_3 + \frac{1}{2} q_2^2 q'_1 + q_1 q_2 q'_2 &= 0 \end{aligned} \right\}, \quad x = -1, \quad (12a)$$

$$q'_3 = 0, \quad x = 1. \quad (12b)$$

Equation (12) written in matrix notation reads:

$$w_1 q' = 0, \quad w_1 = \begin{pmatrix} -\frac{\gamma}{\gamma-1} \frac{q_3}{q_1^2} & q_2 & \frac{\gamma}{\gamma-1} \frac{1}{q_1} \\ \frac{1}{2} q_2^2 & q_1 q_2 & 1 \end{pmatrix}, \quad x = -1, \quad (13)$$

$$w_2 q' = 0, \quad w_2 = (0 \ 0 \ 1), \quad x = 1. \quad (14)$$

This result leaves us with one degree of freedom at the inflow boundary and two degrees of freedom at the outflow boundary. In other words, we are looking for the null spaces to find the missing vectors in order to comply with (7). With $\mathcal{A} \equiv \frac{\partial AF(q)}{\partial q}$, suppose that at $x = -1$

$$w^T \mathcal{A}^T v_{\text{in}} = 0 \quad \forall w \in \text{Null}(w_1). \quad (15)$$

The rank of w_1 is 2 and the kernel has dimension 1:

$$\text{Null}(w_1) = \text{Span}\{w_{1,1}\} = c_1 \begin{pmatrix} 1 \\ -\frac{\gamma}{2q_1 q_2} (2q_3 + q_2^2) \\ -\frac{1}{2}(\gamma-1)q_2^2 + \gamma q_3 \end{pmatrix}, \quad (16)$$

with c_1 a constant.

Likewise, suppose that at $x = 1$

$$w^T \mathcal{A}^T v_{\text{in}} = 0 \quad \forall w \in \text{Null}(w_2). \quad (17)$$

The rank of w_2 is 1 and the kernel has dimension 2:

$$\text{Null}(w_2) = \text{Span}\{w_{2_1}, w_{2_2}\} = c_2 (1 \ 0 \ 0)^T + c_3 (0 \ 1 \ 0)^T, \quad (18)$$

with c_2 and c_3 constants. Multiplying the Null vectors with the Jacobian gives

$$w_{2_1}^T \mathcal{A}^T = (q_2 \ q_2^2 \ \frac{1}{2}q_2^3) \quad (19a)$$

$$w_{2_2}^T \mathcal{A}^T = (q_1 \ 2q_1q_2 \ \frac{\gamma}{\gamma-1}q_3 + \frac{3}{2}q_1q_2^2). \quad (19b)$$

Summarising, the adjoint problem can be seen as a new problem with

$$\mathcal{A}^T \frac{dv}{dx} - \mathcal{B}^T v = \frac{\partial p}{\partial q} \quad (20)$$

in the internal domain and

$$w_L v = 0, \quad w_R v = 0 \quad (21)$$

at the boundaries, where for the subsonic case w_L is a 1×3 matrix and w_R a 2×3 matrix.

4. DISCRETE ADJOINT APPROACH

Whether the residual and output functional are linearised around a given design variable [1] or around a given mesh [5], the Jacobian of the discrete residual written as

$$\int_{\Omega} \frac{\partial Aq}{\partial t} d\Omega = - \left[(AF)_{i+\frac{1}{2}} - (AF)_{i-\frac{1}{2}} \right] + h \frac{dA_i}{dx} J(q_i) = -R_h(q_h) \quad (22)$$

is needed in order to set up and solve the discrete adjoint equations. Derivation of adjoint equations from discrete primal equations has the advantage of not having to worry about adjoint boundary conditions. The Jacobian of the residual operator contains the influence of the primal boundary conditions. When taking the transposed Jacobian for computation of the adjoint solution, the adjoint boundary conditions are automatically included in the system of equations. This can be illustrated in the following way. Writing out the residual operator (22) for volume Ω_i reads

$$A_{i+\frac{1}{2}} \left(\frac{\partial F}{\partial q_i} (q_i, q_{i+1}) q'_i + \frac{\partial F}{\partial q_{i+1}} (q_i, q_{i+1}) q'_{i+1} \right) - A_{i-\frac{1}{2}} \left(\frac{\partial F}{\partial q_{i-1}} (q_{i-1}, q_i) q'_{i-1} + \frac{\partial F}{\partial q_i} (q_{i-1}, q_i) q'_i \right) - \frac{dA_i}{dx} \frac{\partial J}{\partial q_i} (q_i) q'_i = r'_i, \quad (23)$$

where q'_{i-1} , q'_i and q'_{i+1} are the local solution perturbations, and where r'_i is the residual due to linearisation. This equation gives 3 sub-Jacobians around the main diagonal of $\frac{\partial R_i}{\partial q_i}$, denoted by $R_{i,i-1}$, $R_{i,i}$, and $R_{i,i+1}$. At the boundaries, only 2 sub-Jacobians exist, in which $R_{1,1}$ and $R_{N,N}$ contain contributions due to implied boundary conditions, e.g. through the outflow flux

at the right side of the domain: $F_{N+\frac{1}{2}} = F_{N+\frac{1}{2}}(q_N + \text{b.c.})$. The residual operator for volume Ω_N reads

$$A_{N+\frac{1}{2}} \frac{\partial F_R}{\partial q} q'_N - A_{N-\frac{1}{2}} \left(\frac{\partial F}{\partial q_{N-1}}(q_{N-1}, q_N) q'_{N-1} + \frac{\partial F}{\partial q_N}(q_{N-1}, q_N) q'_N \right) - \frac{dA_N}{dx} \frac{\partial J}{\partial q_N}(q_N) q'_N = r'_N \quad (24)$$

However, there is a catch. Strong enforcement of the boundary conditions in the primal residual operator, imply corresponding restrictions for the dual solution space. However, a boundary treatment that yields the correct primal boundary conditions does not automatically yield the correct dual boundary conditions. This difficulty can be avoided by imposing the boundary conditions of the primal residual operator in weak form. The advantage is that no additional restrictions to the solution space are necessary and the resulting dual problem is automatically well-posed problem. This property makes implementation of the numerical adjoint method in a general purpose flow solver attractive. The user of the software takes full advantage of adjoint based grid adaptation, without being burdened by setting up a well-posed dual problem.

To evaluate the adjoint solution in a fully numerical way, the Jacobian matrix $\frac{\partial R_n(q_n)}{\partial q}$ is evaluated with a technique, called Automatic Differentiation or Algorithmic Differentiation (AD) [3]. A common way of obtaining approximate numerical derivatives of a given function is the divided difference approach, but the main disadvantage of obtaining derivatives in this way is that the method suffers from truncation errors and is prone to reduce the number of significant digits by a factor 1.5 or 2. In contrast, AD returns derivatives exact.

5. RESULTS AND CONCLUSIONS

As an example, numerical results for a subsonic channel flow are given. Corresponding dual solutions are shown in figure 1 and 2. Computation of the primal problem with strong boundary conditions leads to significant layers near the boundaries (figure 1). The layers have (almost) disappear when using weak boundary conditions (figure 2). Using different schemes for solving the primal problem, results in different discrete adjoint solutions. For instance, the corresponding primal problems of the dual solutions in figure 2 have been computed with the Lax scheme (left) and the Osher scheme (right).

The main conclusions are listed below:

- In contrast to the numerical adjoint method, the analytical adjoint method requires derivation of adjoint boundary conditions.
- When using the numerical adjoint method, use of weak boundary conditions for the primal problem is advisable in order to prevent erroneous values of the dual solution near the boundaries.
- In contrast to the analytical adjoint method, the numerical adjoint method includes information of the flux evaluation scheme.
- For maximum accuracy, derivatives evaluated by AD are recommended.

For use within general purpose flow solvers, the numerical adjoint method combined with AD techniques is recommended.

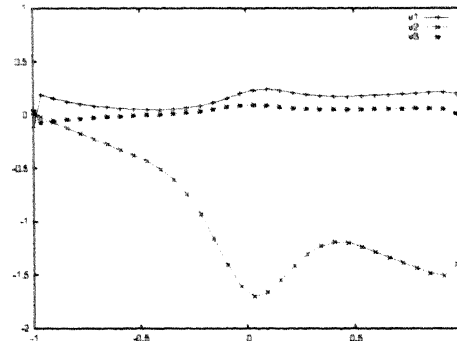


Figure 1. Discrete adjoint solution with explicit boundary conditions, Lax scheme, with ρ_{in} , u_{in} , p_{out} and $h = \frac{1}{16}$.

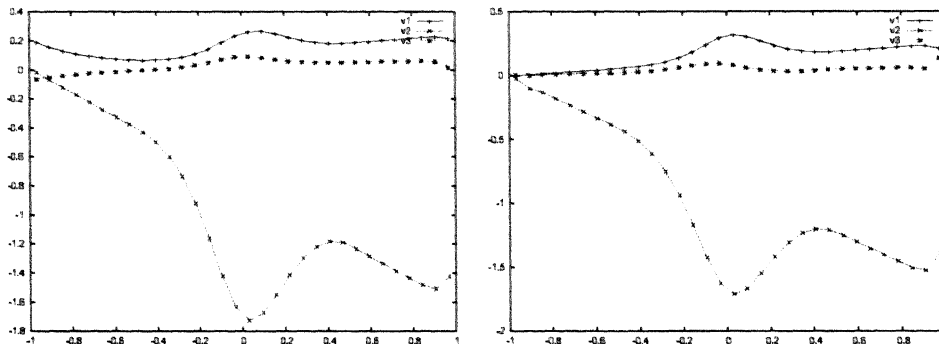


Figure 2. Discrete adjoint solution with implicit boundary conditions for the Lax scheme (left) and Osher scheme (right) with ρ_{in} , u_{in} , p_{out} and $h = \frac{1}{16}$.

REFERENCES

1. Giles MB, Pierce NA, Adjoint equations in CFD: duality, boundary conditions and solution behaviour. *AIAA Paper* 1997; 97-1850.
2. Giles MB, Pierce NA, Analytic adjoint solutions for the quasi-one-dimensional Euler equations, *J. Fluid Mech.* 2001; **426**:327-345.
3. Griewank A, Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation, *SIAM*, Philadelphia, 2000.
4. Koren B, Lewis MR, van Brummelen EH, van Leer B, Riemann-problem and level-set approaches for two-fluid flow computations I. Linearized Godunov scheme, *Report MAS-R0112* 2001; CWI, Amsterdam.
5. Vendetti DA, Darmofal DL, Adjoint error estimation and grid adaptation for functional outputs: application to quasi-one dimensional flow, *J. Comput. Phys* 2000; **164**:204-227.
6. Vendetti DA, Darmofal DL, Grid adaptation for functional output: application to two-dimensional inviscid flows, *J. Comput. Phys* 2002; **176**:40-69.