44th AIAA Aerospace Sciences Meeting and Exhibit, 9-12 Jan., Reno, Nevada

A Newton-Krylov Algorithm with a Loosely-Coupled Turbulence Model for Aerodynamic Flows

Max Blanco* and David W. Zingg†
University of Toronto Institute for Aerospace Studies
4925 Dufferin St., Toronto, Ontario M3H 5T6, Canada

A fast Newton-Krylov algorithm is presented which solves the turbulent Navier-Stokes equations on unstructured two-dimensional grids. The model of Spalart and Allmaras provides the turbulent viscosity and is loosely coupled to the mean flow equations. It is often assumed that the turbulence model must be fully coupled in order to obtain the full benefit of an inexact Newton algorithm. We demonstrate that a loosely-coupled algorithm is effective and has some advantages, such as reduced storage requirements and smoother transient oscillations. A transonic single-element case converges to $1 \cdot 10^{-12}$ in 90 seconds on recent commodity hardware while the lift coefficient is converged to three figures in one quarter of that time.

I. Introduction

The aerodynamics community is interested in rapid convergence to steady-state solutions of the turbulent Navier-Stokes equations. There are several classes of solvers intended for this purpose, one of which is labelled as Newton-Krylov. Recent Newton-Krylov algorithms for flow solutions have fully coupled their turbulence model to the mean flow equations. This paper is meant to explore a loosely coupled algorithm. The authors plan to demonstrate that their loosely-coupled algorithm has performance similar to fully-coupled variants with an attendant reduction in storage costs.

Van Dam *et al.* [1] and Venkatakrishnan [2] seem to have been the first researchers to have implemented Newton algorithms for compressible flows. These made use of variants of Gaussian elimination for the solution of the system of equations. The advent of an algorithm entitled "GMRES: A Generalized Minimum Residual Algorithm for Solving Nonsymmetric Linear Systems", which was proposed in an article by Saad and Shultz [3], allowed the employment of an iterative process instead of the direct solvers used by the aforementioned researchers. Direct solvers suffer from storage and latency costs as compared with iterative solvers when the number of unknowns is large. The quadratic convergence of the Newton-Raphson method is the primary attractor of this type of algorithm, but Venkatakrishnan demonstrated in his seminal work [2] that entry into the region of convergence might be delayed: a transonic Euler case required roughly 40 iterations prior to rapid convergence. Similar results were shown by Orkwis and McRae [4] and Vanden and Orkwis [5] for hypersonic laminar cases, Forsyth and Jiang [6] for sub- and supersonic laminar cases, and by Barth and Linton [7] for a high-lift configuration.

*Doctoral Candidate. Student Member, AIAA.
†Corresponding Author. Professor, Senior Canada Research Chair in Computational Aerodynamics, 2004 Guggenheim Fellow, Senior AIAA Member.
Copyright © 2006 by Max Blanco and David W. Zingg. Published by the American Institute of Aeronautics and Astronautics, Inc. with permission.

The turbulence model employed in the present study is due to Spalart and Allmaras [8, 9]. Barth and Linton [7] employed this model in a fully-coupled form. Anderson *et al.* [10] did not investigate their loosely-coupled model within the context of their Newton-Raphson method for incompressible flows, as they wrote that quadratic convergence rates are unattainable with such a method. Geuzaine [11] implemented the same model in a fully-coupled form, as well as a $k-\omega$ model, again in fully-coupled form. Chisholm and Zingg [12] presented a fully-coupled model; their study was performed on structured grids. Smith *et al.* [13] also employ a fully-coupled turbulence model.

The purpose of this work is to demonstrate the flexibility of loosely-coupled algorithms, the reduction in storage costs as compared to fully-coupled schemes, and that the loosely-coupled approach can be as fast or faster to converge than fully-coupled algorithms. The remainder of this work is structured as follows: we describe the algorithm in §II, which is then compared to experimental data in §III. We choose and study a metric of comparison in §IV, and discuss the optimization of the algorithm in §V. Section VI demonstrates the capabilities of the algorithm. Section VII is concerned with comparisons to fully-coupled algorithms. A summary completes the work in §VIII. The nomenclature is added as an appendix, and a table of test cases and relevant data along with labels employed throughout this study completes the paper.

II. Algorithm Description

2.1. Newton-Raphson Method

Our mathematical model is formed by a boundary value problem (BVP). We employ a grid composed of triangles to discretize the governing equations, thus to form a residual vector, denoted $\mathbf{R}(\mathbf{Q})$, where $\mathbf{Q}$ is the solution vector. We have employed bold type to denote vector quantities throughout this work. Each grid node, subscript i in the general case, is associated with a solution vector. The method applied for the solution of the system of equations is due to Newton and Raphson. We employ a Fréchet finite difference to solve the problem in matrix-free form.

We write the solution process as

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right)^k \cdot \Delta \mathbf{Q}^k = -\mathbf{R}\left(\mathbf{Q}^k\right) \tag{1}$$

solve this equation for $\Delta \mathbf{Q}^k$. The index superscript k labels the "outer" iteration. The solution to Eq. 1 is accomplished by another iterative process, due to Saad and Schultz [3], named GMRES. The GMRES process allows us to solve Eq. 1 by means of successive approximation given by a first-order Fréchet difference,

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right) \cdot \mathbf{v}^l = \frac{\mathbf{R}\left(\mathbf{Q} + \epsilon^l \mathbf{v}^l\right) - \mathbf{R}\left(\mathbf{Q}\right)}{\epsilon^l} \tag{2}$$

The differentiation process involves a scalar parameter ϵ, which we choose as

$$\epsilon^l = \frac{\sqrt{\epsilon_{mz}}}{||\mathbf{v}^l||_2} \tag{3}$$

where ϵ_{mz} depends on the precision with which the particular machine represents floating point numbers.

Once the GMRES, or "inner", iterations labelled l terminate (see our previous article [14] for implementation details), we update the solution vector $\mathbf{Q}^{k+1} = \mathbf{Q}^k + \Delta \mathbf{Q}^k$. Equations 1 to 3 constitute the algorithm to which many in this community refer as matrix-, or Jacobian-free. The present authors [14] found that a permutation due to Cuthill and McKee [15] with modifications by Liu and Sherman [16] and Gibbs *et al.* [17] (see [17] §4.1, Algorithm I) implemented by Balay *et al.* [18] performs well when combined with the particular preconditioner. This preconditioner is known as BILU(n), and formed by a process with roots in Meijerink and van der Vorst [19]. Our previous studies showed that the block-factored BILU(n) outperforms

the scalar-factored ILU(n) by a significant margin, and block-scalar ILU(n), which we denoted B/SILU(n), by a less significant though still notable margin.

We solve the Navier-Stokes equations in conservative variable form, $\mathbf{Q}_i = [(\rho)_i, (\rho u)_i, (\rho v)_i, (e)_i, (\rho\nu)_i]^T$. The fundamental difference between the present loosely-coupled scheme and fully-coupled schemes is the separation of variables. Fully-coupled schemes form vectors such that the Jacobian block dimension is of five in two dimensions. We write our solution as two distinct processes, both of which employ the same Newton-Raphson algorithm, to solve the mean flow, ${}_1\mathbf{Q}_i = [(\rho)_i, (\rho u)_i, (\rho v)_i, (e)_i]^T$, independently from the turbulent flow, ${}_2\mathbf{Q}_i = [(\rho\nu)_i]^T$. The Jacobian matrix blocks then are four-dimensional and one-dimensional, respectively. The loose coupling of the present scheme means that the inner iterations of the mean flow equations are completed, then the inner iterations of the turbulence model are completed, before the outer iteration index is advanced by one.

2.2. Navier-Stokes Equations

The integral form of the steady-state equations of motion of a compressible viscous fluid, with suitable non-dimensionalization, may be written in tensor notation for an arbitrary control volume labelled $\partial\Omega$ as

$$\mathbf{R}(\mathbf{Q}) \equiv \oint_{\partial\Omega} (\mathbf{F}\cdot\hat{\mathbf{n}}\,\mathrm{d}S - \mathbf{G}\cdot\hat{\mathbf{n}}\,\mathrm{d}S) = \mathbf{0} \tag{4}$$

where $\hat{\mathbf{n}}$ is the unit normal, $\mathbf{F}$ is the inviscid flux tensor,

$$\mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (e+p)\,u \end{bmatrix} \hat{\mathbf{i}} + \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ (e+p)\,v \end{bmatrix} \hat{\mathbf{j}} \tag{5}$$

and $\mathbf{G}$ the flux tensor associated with viscosity and heat conduction. We assume Stokes' hypothesis holds for air, which behaves in our régime like a Newtonian fluid:

$$\mathbf{G} = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} - q_x \end{bmatrix} \hat{\mathbf{i}} + \begin{bmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ u\tau_{yx} + v\tau_{yy} - q_y \end{bmatrix} \hat{\mathbf{j}} \tag{6}$$

where single subscripts x or y denote partial differentiation with respect to the coordinate, and stress terms have been denoted by τ. Two groups of symbols in Eq. 6 remain to be defined in the next two paragraphs: $\mathbf{q}$ and τ, with the aid of the Boussinesq and Reynolds analogies. The former analogy is employed to write terms similar to those which involve the dynamic viscosity in order to construct a dynamic turbulent viscosity, which is addressed in the next subsection. The latter analogy serves to apply the turbulent viscosity to the problem of heat transfer.

The viscous stress tensor τ can be related to viscosity and the strain rate tensor as

$$\tau = (\mu+\mu_t)\begin{bmatrix} 2u_x & u_y+v_x \\ v_x+u_y & 2v_y \end{bmatrix} - \frac{2}{3}(\mu+\mu_t)\begin{bmatrix} u_x+v_y & 0 \\ 0 & u_x+v_y \end{bmatrix} \tag{7}$$

in which dynamic viscosity depends on temperature. We follow Schlichting [20] * and Mavriplis [21] to write a power law relation to model this feature

$$\frac{\mu}{\mu_\infty} = \left(\frac{T}{T_\infty}\right)^{0.71} \tag{8}$$

*See Schlichting, figure 15.1, page 340.

The heat flux vector, $\mathbf{q}$, is given by Fourier's law, augmented by appeal to the Reynolds analogy,

$$\mathbf{q} = -\frac{1}{\gamma - 1}\left[\frac{\mu}{\mathcal{P}r} + \frac{\mu_t}{\mathcal{P}r_t}\right]\nabla T \tag{9}$$

The term which involves dynamic viscosity is divided by the Prandtl number, represented by the symbol $\mathcal{P}r \equiv 0.72$. The terms subscripted with t will be treated in the next section. A thermally and calorically perfect gas is assumed, giving

$$p = (\gamma - 1)\left[e - \frac{\rho}{2}(u^2 + v^2)\right] \tag{10}$$

with $\gamma = 1.4$. We follow Mavriplis [21] insofar as the integration of convective and viscous fluxes is concerned: the convective fluxes are treated as piecewise linear functions in space, while the viscous fluxes are piecewise constant over each triangle, since they are formed from first derivatives of flow variables.

A Riemann-type scheme as detailed by Thomas and Salas [22] is employed at the farfield boundary. We require that the velocity be zero at a solid wall, that the heat flux be adiabatic and that the normal gradient of pressure be zero [23].

2.3. Turbulence Model

The Navier-Stokes equations are augmented with a turbulence model due to Spalart and Allmaras [8, 9]. This model has been chosen here for the following reasons: its accuracy, its applicability to unstructured grids, and its economy relative to models with two or more equations. The turbulence model simulates an eddy viscosity, ν_t, through convective, diffusive and productive terms. We write the equation

$$\mu_t \equiv \rho\bar{\nu}\frac{\left(\frac{\bar{\nu}}{\nu}\right)^3}{\left(\frac{\bar{\nu}}{\nu}\right)^3 + c_{v1}^3} \tag{11}$$

which then allows the variable μ_t to be incorporated into the mean flow as an additional dynamic viscosity and, with division by $\mathcal{P}r_t \equiv 0.90$ [20, 9] †, as an additional heat flux term. A BVP is solved for the working variable $\bar{\nu}$. We have implemented the model within the framework of the finite-volume scheme applied to our triangular grids. The turbulent production, destruction and trip terms, although included in the residual operator, are not linearized in our preconditioner. We have yet to experiment with the updated trip control constants c_{t3} and c_{t4} documented in the article [9], and prefer instead to employ those values listed in reference [8]. Values of $\bar{\nu}$ in the freestream on the farfield boundary are set to $\nu_\infty = 0.01$, and are zero at solid surfaces.

The turbulence model is iterated separately from the mean flow. This allows the evolution of each model to proceed along distinct paths, which is of benefit because

- each system makes appeal to different physical arguments;
- each system has its own numerical properties;
- investment is reduced in case a change in model is desired.

2.4. Grid Sequencing

Most researchers who have reported Newton-Raphson-like algorithms employ grid sequencing (first demonstrated to great effect by Venkatakrishnan [2]). This strategy serves to dampen initial transients, which are especially notable in transonic Euler flows, as well as to accelerate the convergence. Grid sequencing has been added to the present algorithm in an effort to speed convergence. The grids employed here were not created by nesting or agglomeration from the finest grid, nor was the fine grid created by adaptation, all of

† See Schlichting, §19.g.

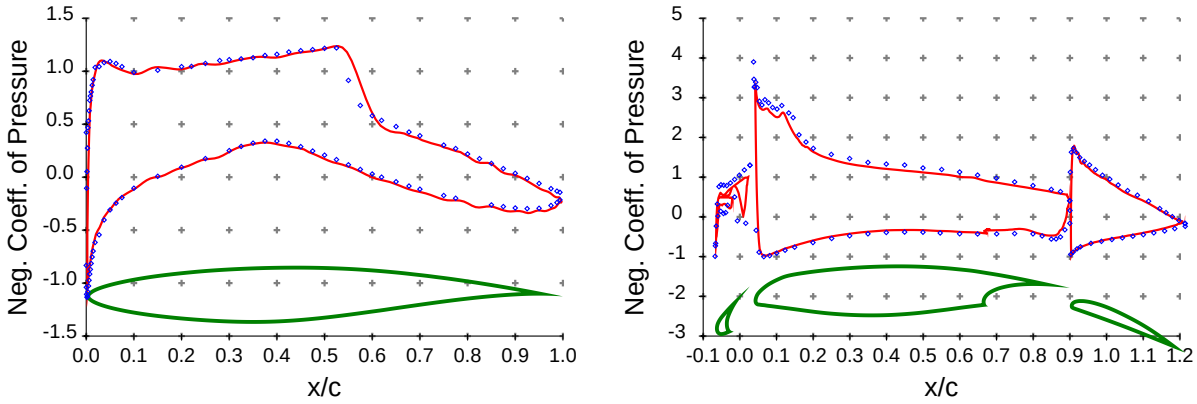

Figure 1. Calculated surface coefficients of pressure compared with overlays of those measured by (a) Cook *et al.* [26] and (b) Moir [27].

which have been employed with success elsewhere. The grids that were employed were created independently in a pre-processing step. We employ a sequence of grids in which the coarsest grid contains approximately $1 \cdot 10^3$ points, and a loose geometric progression with ratio somewhere between three and four serves to indicate the number of points contained in the subsequent grid.

2.5. Initial Iterations

We noted in our previous work [14] that during the initial iterations of complex transonic flows, the second order Jacobian method might diverge, and an approximate Jacobian matrix was preferred. Initial iterations seem to be a problem with most—if not all—fully-coupled Newton-Krylov algorithms reported to date. The start-up strategies reported in the literature, usually some sort of pseudo-time integration, can be quite complex and require many parameters to calm oscillations and allow convergence. The algorithm described in this paper needs no such device when grid sequencing is employed.

III. Comparison to Experiment

Aerodynamic prediction algorithms require testing against experimental data to ensure accuracy, and to identify areas of possible improvement. Two collections of data [24, 25] have been gathered and made available to scientists by a North Atlantic Treaty Organization subcommittee called the Advisory Group for Aerodynamic Research and Design (AGARD). This body was charged with the distillation of a wide variety of experimental data to those with flow features of interest to present-day aerodynamicists. We have chosen to present two series of experimental data, one due to Cook *et al.* [26], the other due to Moir [27], which we precede with a discussion of the computational grids used.

3.1. Test Cases & Grids

The airfoils are represented by coordinate pairs which we place on natural cubic splines. The external farfield boundary is represented by a ring of points that approximate a circle, although the shape of this boundary is unimportant and effectively arbitrary.

A grid triangulator due to Walsh and Zingg [28] is employed for viscous flows. They encode an advancing front method and Minmax triangulation surveyed by Barth [29]. The advancing front method allows the specification of node positions by any criterion, which is important due to the great variance in streamwise

Label	Case	$\mathcal{M}_\infty$	α	$\mathcal{R}e$	Offwall	Nodes	foil/farf
WZ3C [26]	Airfoil RAE	0.729	2.31	$6.50 \cdot 10^6$	$2 \cdot 10^{-6}$	16,948	300/72
L1T2 [27]	NLHP 3elem	0.197	4.01	$3.52 \cdot 10^6$	$2.5 \cdot 10^{-6}$	51,779	1376/82

Table 1. Parameters for comparison to experiment of Figure 1.

Processor	Clock
Macintosh 750	267 MHz
Pentium III	450 MHz
Pentium III	600 MHz
Athlon XP	1800 MHz
Athlon XP	2167 MHz

Table 2. Machinery used for test of present algorithm.

and normal gradients typical of viscous flows. Walsh and Zingg followed Barth [30] when they employed a stretched Steiner criterion to generate the placement of nodes in the near-foil region, outside of which they employed an algorithm due to Weatherill [31] to fill the remainder of the computational domain.

3.2. Comparison to Experiment

We repeat with our algorithm two experimental tests, summary data of which can be found in Table 1. We compare algorithmic predictions with the corresponding experimental data to determine whether our predictor is accurate. The experimental data of Cook *et al.* [26] document tests performed on a rear-loaded subcritical airfoil with a thickness to chord ratio of 12.1% given the name RAE 2822. A three-element airfoil data set was published as chapter A/2 in AGARD Advisory Report #303 [27]. This airfoil was studied under the U. K. National High Lift Programme (NLHP) around 1970.

The coefficient of pressure on the surface of airfoil RAE 2822 is shown in Figure 1(a). Case 6 of Cook *et al.* had nominal Mach number 0.725, angle of attack 2.92. We calculate the flow with Mach number 0.729, angle of attack 2.31, as most other calculational literature. The algorithm produces good agreement with the experimental data. The coefficient of pressure on the surface of the three-element configuration is reproduced as Figure 1(b). The grid was comprised of 51,779 nodes, and produces good agreement with experimental data over all three elements. In both cases, the agreement with experiment is comparable to that obtained by other authors on similar grids.

IV. Characterization of Metric of Comparison

Comparison between algorithms would be accomplished more easily if all computations were performed on the same machine and the same grid. The elapsed time for completion would then be the metric of choice, however various factors render the conditions under which this type of comparison could occur extremely unlikely. A metric based on equivalent residual function evaluations (also known as right-hand-side evaluations), denoted as "Eq. $\mathbf{R}(\mathbf{Q})$" in tabular form, should transcend machinery and makes algorithm comparison somewhat possible. Pueyo and Zingg [32] remark one of its weaknesses: "...it tends to favour expensive flux evaluation methods [because] overhead appears smaller...", but it was the only cross-platform measure they found acceptable for their evaluations. We choose to employ the same metric to evaluate the

Label	Eq $\mathbf{R}(\mathbf{Q})$	σ
PZ1	360	50
PZ2	660	100
PZ3	550	140
PZ4	625	25
PZ5	780	70
PZ6	860	120
PZ7	2,020	430
PZ8	850	170

Table 3. Equivalent residual evaluations for a range of cases evaluated on five processors (see Table 2).

performance of our algorithm. We reduce the L_2-norm of the residual to 10^{-12} and note that lift and drag coefficients are converged to three significant figures in roughly one quarter of the time to reduce the residual to 10^{-12}.

The present algorithm was was ported onto different 32-bit machines, and reduced to machine language with similar GNU 'C' and Fortran compilers, for all of which we employed similar compiler flags. Table 2 documents the type and clock frequency of the machinery on which the algorithm was tested.

The number of equivalent function evaluations is processor dependent. The test data are characterized to follow a Gaussian distribution across machines, with the use of parameters for the mean and standard deviation, σ. The standard deviation is typically on the order of 20% of the mean, as summarized in Table 3. We have assumed implicitly in our use of the Gaussian distribution that the data are random. Although this is not the case, the parameter σ provides useful information about the scatter or variation of the data. The two Intel machines tested report a consistently lower $\mathbf{R}(\mathbf{Q})$ score than the two AMD machines tested. All subsequent data are reported for an AMD Athlon XP which clocked 2167 Mhz.

V. Optimization of Loosely-Coupled Algorithm

5.1. Turbulence Model ILU(n)

Geuzaine *et al.* [11] report the continuity residual separately from the turbulence model residual, as do Chisholm and Zingg [33], however it should be noted that the full coupling of their algorithms fixes the relation between residual quantities. The present algorithm adds a degree of freedom by its nature. We define for every outer iteration two iterative processes: the mean flow process and the turbulent flow process. The mean flow is solved first and thereafter alternates with the turbulence model iteration as reported in §2.1.

For each mean flow iteration, we perform a number of turbulence model iterations such that the turbulent residual is less than the mean flow residual at every outer iteration. This is demonstrated in Figure 2(a). The fully-coupled algorithms with which we contrast here have no such freedom. The fully-coupled models for which we have seen statistics report turbulent residuals consistently greater than the mean flow residuals.

The loosely-coupled algorithm permits us to choose different fill levels for the mean flow (BILU(n)) and turbulence model (ILU(n)) iterations. Single element cases typically are solved for optimum wallclock performance with BILU(4), while multi-element cases may require one or two more levels of fill.

Table 4 and Fig. 2(b) present data for an optimization study of the fill level for the turbulence model. The mean flow matrix problem, whose storage requirements in Megabytes are indicated by "MF", remains with BILU(4) as its preconditioner. The ILU preconditioner for the turbulence model was varied between ILU(0) and ILU(9). Its storage requirement is indicated by "TM". The mean flow solver required upwards

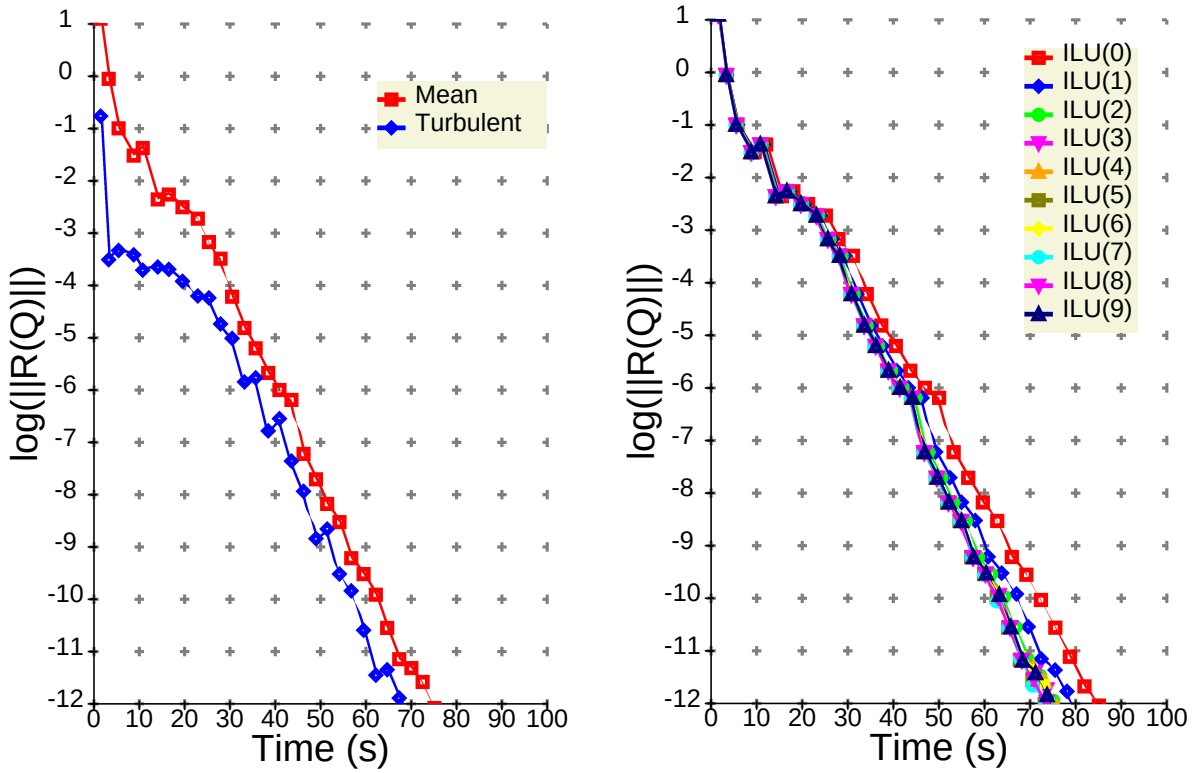

Figure 2. Convergence histories for case PZ8. Left hand side depicts mean flow residual and turbulence model residual for optimal solution. Effect of turbulence model ILU fill parameter on overall solution time to the right, from which the turbulence model residual has been omitted for clarity.

of 500 inner iterations to converge to a residual of $1 \cdot 10^{-12}$. Net inner iterations for the turbulence model are shown in the table as Σin. Wallclock time is minimized for $n = 5$ or $n = 7$. However, for this problem we are more interested in a tradeoff between storage requirements and performance, and thus we choose as turbulence model preconditioners a range between ILU(2) and ILU(4).

5.2. Grid Sequencing

We find our algorithm to converge most quickly if the sequencing switch is set to $1 \cdot 10^{-2}$, as shown in Table 5. Geuzaine [11] converges to near machine zero on his coarser grids, whereas Pueyo and Zingg [34] converge to $1 \cdot 10^{-2}$ on theirs. Chisholm and Zingg [12] have a switch set nearer to $1 \cdot 10^{-5}$.

The results presented in Table 6 indicate the benefit of sequencing for single-element cases, and those in Table 7 for a multi-element foil. The benefit is much greater for the latter case, perhaps due to the more complex flow field.

VI. Calculation of a Multi-element Airfoil Flow

A take-off geometry from the United Kingdom National High Lift Programme is depicted in Figure 1(b). The wind tunnel measurement data were published in 1994 by the AGARD as chapter A/2 of Advisory Report #303 [27]. The slat, denoted "L1" by the experimenters [35], and flap, denoted "T2", is examined

ILU(n)	MF	TM	Σin	Time (s)
0	37.9	1.3	551	85
1	37.9	1.6	350	81
2	37.9	2.2	249	78
3	37.9	2.8	209	76
4	37.9	3.4	196	76
5	37.9	4.0	183	73
6	37.9	4.7	174	76
7	37.9	5.3	162	73
8	37.9	5.9	167	76
9	37.9	6.5	160	76

Table 4. Results of turbulence model ILU(n) parameter study for 13,514-node grid about airfoil RAE 2822. The storage requirements for the mean flow, indicated by MF, and the turbulence model, indicated by TM, are measured in Megabytes (Mb). Convergence was defined at $L_2(\mathbf{R}) = 1 \cdot 10^{-12}$.

Sequence at	Time (s)
$1 \cdot 10^{-1}$	76
$1 \cdot 10^{-2}$	63
$1 \cdot 10^{-3}$	68
$1 \cdot 10^{-4}$	69
$1 \cdot 10^{-5}$	70
$1 \cdot 10^{-6}$	71
$1 \cdot 10^{-7}$	72
$1 \cdot 10^{-8}$	72
$1 \cdot 10^{-9}$	73
$1 \cdot 10^{-10}$	74

Table 5. Residual at which to sequence for case PZ8.

for the case in which the airfoil was set at an angle of attack of 4.01^o to the oncoming flow. Fejtek reported a group of results [36, 37] for this case. The free stream Mach number was $\mathcal{M}_\infty = 0.197$, while the Reynolds number calculated to $\mathcal{R}e = 2.51 \cdot 10^6$ based on the nested configuration chord length, c. The leading-edge slat, positioned at an angle of 25^o, measured $12.5\%c$, while the flap measured $33\%c$ and was positioned with a deflection angle of 20^o.

The results in Table 7 are summarized from a study of convergence data. The baseline GMRES vector subspace was GMRES(50,0) and the preconditioner was set to BILU(4). We were able to increase the rate of convergence by almost a factor of two, while storage requirements only increased by 25% when we increased the fill level to BILU(6) and employed a sequence of three grids for this complex flow. The surface coefficient of pressure is depicted in Figure 1(b). We can solve this flow with a Pentium class computer in under ten minutes.

Case	#grids	Time (s)	$\mathbf{R}(\mathbf{Q})$
PZ5	1	121.8	1,220
PZ5	2	112.1	1,130
PZ5	3	110.2	1,100
CZ2	1	91.6	1,150
CZ2	2	83.8	1,050
CZ2	3	89.9	1,130

Table 6. Grid sequencing study: number of grids, time to convergence, and equivalent residual evaluations for cases PZ5 and CZ2 to reduce the residual to $1 \cdot 10^{-12}$.

#grids	$\mathbf{R}(\mathbf{Q})$	Time (s)	GMRES	BILU(n)	Storage (Mb)
L1T2 : 1	4970	1,340	(50,0)	4	328
L1T2 : 3	2048	530	(50,0)	6	388

Table 7. Case L1T2: Equivalent residual evaluations and other statistics for 10 order of magnitude residual reduction.

Dimensions	2		3	
Model	Hexagon		Icosahedron	
Coupling	Full	Loose	Full	Loose
Average neighbours per node	6	6	20	20
Storage per block	5*5	4*4+1	6*6	5*5+1
Total per block equation	175	119	756	546
Economy		32%		28%

Table 8. Generalized storage requirements: a comparison between full and loose coupling in two and three dimensions.

VII. Comparison with Fully-Coupled Schemes

7.1. Memory Advantages

The distinction between loosely- and fully-coupled turbulence models is shown in Table 8 for a hexagonal node layout, which serves as a model for our two-dimensional grid creation algorithm. An icosahedral extrapolation for three-dimensional grids is tabulated as well. The loosely coupled approach leads to a reduction in storage requirements of the order of 30% in both two and three dimensions.

7.2. Equivalent R(Q) Comparison

We compare in Table 9 our scheme to various others which have appeared in the literature. Data are included for Geuzaine [11], Wong and Zingg [38] and Chisholm and Zingg [12]. In each comparison we replicated as well as we could the number of nodes, offwall spacing, and other relevant and published mesh parameters. The summary data is appended as Table 10.

						Eq. $\mathbf{R}(\mathbf{Q})$			
Label	Foil	$\mathcal{M}_\infty$	α	$\mathcal{R}e \cdot 10^6$	Grids	Present	Chisholm & Zingg [12]	Geuzaine [11]	Wong & Zingg [38]
G1	NAC	0.502	1.77	2.91	3	630	—	1,500	—
CZ1	NAC	0.300	6.00	9.	3	900	900	—	—
WZ3C	RAE	0.729	2.31	6.5	1	1,300	—	—	3,000
CZ2	RAE	0.729	2.31	6.5	3	1,120	900	—	—
G3	NLR	0.185	6.00	2.51	1	1,220	—	10,000	—
CZ3	NLR	0.185	6.00	2.51	3	1,450	1,500	—	—

Table 9. Equivalent residual evaluations for four Newton-Krylov solvers. Number of nodes and offwall spacing of grid, as well as convergence criteria, from each prior case was equalled. References and further grid information may be found in Table 10.

The ability to make comparisons independent of processors is a strength of the metric we have employed throughout this paper. The number of grids employed by the other authors is listed in the sixth column of Table 9, and in each case we employed the same number or less. The CPU time for L_2-norm convergence to $1 \cdot 10^{-12}$ is included in the text. A short paragraph on each of the six lines in the table follows.

Case G1: Geuzaine [11] reports a result from Thibert *et al.* [39], labelled here as case G1, for his fully-coupled implementation of the turbulence model due to Spalart and Allmaras. We note that his algorithm makes use of nested hybrid Cartesian grids, generated by a full-coarsening procedure. His finest grid of three has an offwall spacing of 30 microchords [40], contains 14,110 cells, and requires 2.5 times as many function evaluations as our loosely-coupled algorithm. For this case, the present algorithm converged to 10^{-12} in 25 seconds.

Case CZ1: Chisholm and Zingg [12] report the same airfoil but different flow conditions. Their final grid has an offwall spacing of one microchord and contains 17,385 nodes. Their algorithm employs structured grids. In this case, their fully-coupled and our loosely-coupled algorithm perform similarly. Our loosely coupled algorithm converged in 82 seconds.

Case WZ3C: Wong and Zingg [38] report an algorithm designed to solve on grids comprised of arbitrarily shaped control volumes. We show their results for a grid composed of triangular elements. Their grid has an offwall spacing of one microchord and contains 15,900 nodes. Their fully-coupled algorithm requires roughly 2.3 times as many function evaluations as ours. The time to convergence of our algorithm measured 117 seconds.

Case CZ2: Chisholm and Zingg [12] solve the same flow with a sequence of three grids. Their final grid has 14,619 nodes and an offwall spacing of two microchords. Their algorithm outperforms ours by a factor of approximately 1.3, which is slightly outside the standard deviation of §IV. The time to convergence of our loosely-coupled algorithm is 91 seconds.

Case G3: Geuzaine [11] employs a 19,482 cell grid for this case, which is a calculation of an experiment performed by van den Berg [41]. The grid he employs has an offwall spacing of 20 microchords [40]. He converges his turbulence model three orders of magnitude higher than we do. His density residual converges to $1 \cdot 10^{-10}$ for this case. We converge the mean flow residual to $1 \cdot 10^{-10}$, and our turbulence model lower than that. His algorithm requires over eight times as many function evaluations as ours. Our algorithm converges in 86 seconds.

Case CZ3: Chisholm and Zingg [12] employ 44,059 nodes for the same case. Their grid has an offwall spacing of one microchord. Their algorithm requires the same number (within the standard deviation) of function evaluations as ours. Our time to convergence measured 309 seconds.

VIII. Discussion and Conclusions

A metric employed to compare algorithms across machinery has been characterized. Five processors were tested using the present algorithm and the equivalent function evaluation metric was summarized as a mean and a variation. The variation was on the order of 20%. Put another way, the speed at which the present algorithm runs was determined within a margin of 20%. This measure does not account for differences in implementation, language, or programming skill, but it does allow a universal basis of comparison between algorithms independent of processor.

An original Newton-Krylov solver with a loosely-coupled turbulence model for aerodynamic flows has been presented. The algorithm has been shown to compare well with algorithms that employ fully-coupled turbulence models. The benefits of this loosely-coupled algorithm have been demonstrated via parameter studies of a number of features which distinguish it from fully-coupled variants. The conclusions are enumerated here:

1. The present loosely-coupled algorithm requires **substantially fewer equivalent function evaluations** than the fully-coupled unstructured or hybrid grid algorithms with which it has been compared. (§7.2)

2. The present unstructured loosely-coupled algorithm requires **a similar number of equivalent function evaluations** to a structured fully-coupled algorithm. (§7.2)

3. The present loosely-coupled algorithm requires **30% less memory** than a fully-coupled algorithm. (§7.1)

4. The loosely-coupled approach **reduces investment** in case a new turbulence model is needed.

IX. Acknowledgments

The funding of the first author by the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged, as is the funding of the second author by the Natural Sciences and Engineering Research Council of Canada and the Canada Research Chairs program.

References

[1] van Dam, C. P., M. Hafez and J. Ahmad, "Calculations of viscous flow with separation using Newton's method and direct solver," AIAA Journal, Vol. 28, No. 5, 1990, pp. 937–939.

[2] Venkatakrishnan, V., "Newton solution of inviscid and viscous problems," AIAA Journal, Vol. 27, No. 7, 1989, pp. 885–891. See also AIAA Paper 88-813.

[3] Saad, Y. and M. H. Schultz, "GMRES: A generalized minimum residual algorithm for solving non-symmetric linear systems," SIAM J. Sci. Stat. Computing, Vol. 7, No. 3, 1986, pp. 856–869.

[4] Orkwis, P. D. and D. S. McRae, "Newton's method solver for high-speed viscous separated flow fields," AIAA Journal, Vol. 30, No. 1, 1992, pp. 78–85.

[5] Vanden, K. J. and P. D. Orkwis, "A comparison of numerical and analytical Jacobians," AIAA Journal, Vol. 34, No. 6, 1996, pp. 1125–1130.

[6] Forsyth, P. A. and H. Jiang, "Nonlinear iteration methods for high speed laminar compressible Navier-Stokes equations," Computers and Fluids, Vol. 26, No. 3, 1997, pp. 249–268.

[7] Barth, T. J. and S. W. Linton, "An unstructured mesh Newton solver for compressible fluid flow and its parallel implementation," AIAA Paper 95-0221, 1995.

[8] Spalart, P. R. and S. R. Allmaras, "A one-equation turbulence model for aerodynamic flows," AIAA Paper 92-0439, 1992.

[9] Spalart, P. R. and S. R. Allmaras, "A one-equation turbulence model for aerodynamic flows," in *La Recherche Aérospatiale*, no. 1, pp. 5–21, Chatillon, France: ONERA, 1994.

[10] Anderson, W. K., R. D. Rausch and D. L. Bonhaus, "Implicit/multigrid algorithms for incompressible turbulent flows on unstructured grids," Journal of Computational Physics, Vol. 128, No. 2, 1996, pp. 391–408. See also AIAA Paper 95-1740.

[11] Geuzaine, P., "Newton-Krylov strategy for compressible turbulent flows on unstructured meshes," AIAA Journal, Vol. 39, No. 3, 2001, pp. 528–531. See also AIAA Paper 99-3341.

[12] Chisholm, T. T. and D. W. Zingg, "Start-up issues in a Newton-Krylov algorithm for turbulent aerodynamic flows," AIAA Paper 2003-3708, June 2003.

[13] Smith, T., R. Hooper, C. Ober, A. Lorber and J. Shadid, "Comparison of operators for Newton-Krylov method for solving compressible flows on unstructured meshes," AIAA Paper 2004-743, 2004.

[14] Blanco, M. and D. W. Zingg, "Fast Newton-Krylov method for unstructured grids," AIAA Journal, Vol. 36, No. 4, 1998, pp. 607–612.

[15] Cuthill, E. and J. McKee, "Reducing the bandwidth of sparse symmetric matrices," in *Proc. 24th Nat. Conf. ACM*, pp. 157–172, 1969.

[16] Liu, W.-H. and A. H. Sherman, "Comparative analysis of the Cuthill-McKee and the reverse Cuthill-McKee ordering algorithms for sparse matrices," SIAM Journal of Numerical Analysis, Vol. 13, No. 2, 1976.

[17] Gibbs, N. E., W. G. Poole, Jr. and P. R. Stockmeyer, "An algorithm for reducing the bandwidth and profile of a sparse matrix," SIAM J. Num. Anal., Vol. 13, 1976, pp. 236–249.

[18] Balay, S., W. Gropp, L. C. McInnes and B. Smith, "PETSc 2.0 users' manual," Argonne National Laboratory ANL-95/11, 1998. Revision 2.0.22.

[19] Meijerink, J. A. and H. A. van der Vorst, "An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix," Mathematics of Computation, Vol. 31, No. 148, 1977.

[20] Schlichting, H., *Boundary Layer Theory*. New York: McGraw-Hill Book Company, Inc., 4 ed., 1960.

[21] Mavriplis, D. J., "Euler and Navier–Stokes computations for two–dimensional geometries using unstructured meshes," NASA CR-181977, Jan. 1990.

[22] Thomas, J. L. and M. D. Salas, "Far–field boundary conditions for transonic lifting solutions to the Euler equations," AIAA Journal, Vol. 24, 1986, pp. 1074–1080.

[23] Hirsch, C., *Numerical Computation of Internal and External Flows*, Vol. 2. New York: John Wiley & Sons, 1990.

[24] Various, "Experimental data base for computer program assessment: Report of fluid dynamics panel working group 04,"AGARD Advisory Report #138. Neuilly-Sur-Seine, France: North Atlantic Treaty Organization, May 1979.

[25] Various, "A selection of experimental test cases for the validation of CFD codes,"AGARD Advisory Report #303. Neuilly-Sur-Seine, France: North Atlantic Treaty Organization, August 1994. 2 volumes.

[26] Cook, P. H., M. A. McDonald, and M. C. P. Firmin, "Airfoil RAE 2822 - pressure distributions, and boundary layer wake measurement," in *Experimental Data Base for Computer Program Assessment: Report of the Fluid Dynamics Panel Working Group 04* (J. Barche, ed.), chapter A/6 in AGARD Advisory Report #138. Neuilly-Sur-Seine, France: North Atlantic Treaty Organization, May 1979.

[27] Moir, I. R. M., "Measurements on a two-dimensional aerofoil with high-lift devices," in *A Selection of Experimental Test Cases for the Validation of CFD Codes* (M. Burt, ed.), Neuilly-Sur-Seine, France: North Atlantic Treaty Organization, Aug. 1994.

[28] Walsh, P. C. and D. W. Zingg, "On the accuracy of viscous airfoil computations using solution-adaptive unstructured grids," AIAA Paper 97-329, Jan. 1997.

[29] Barth, T. J., "Aspects of unstructured grids and finite-volume solvers for the Euler and Navier-Stokes equations," 1992. Lectures delivered at the von Karman Institute, Rhode-St. Genèse, Belgium.

[30] Barth, T. J., "Steiner triangulation for isotropic and stretched elements," AIAA Paper 95-213, Jan. 1995.

[31] Weatherill, N. P., "A method for generating irregular computational grids in multiply connected planar domains," International Journal for Numerical Methods in Fluids, Vol. 8, 1988, pp. 181–197.

[32] Pueyo, A. and D. W. Zingg, "An efficient Newton-GMRES solver for aerodynamic computations," AIAA Paper 97-1955, 1997.

[33] Chisholm, T. T. and D. W. Zingg, "A fully coupled Newton-Krylov solver for turbulent aerodynamic flows," in *Proceedings of the 2002 Congress of the International Council of Aeronautical Sciences*, pp. 333.1–333.9, 2002.

[34] Pueyo, A. and D. W. Zingg, "Efficient Newton-Krylov solver for aerodynamic computations," AIAA Journal, Vol. 36, No. 11, 1998, pp. 1991–1997.

[35] Woodward, D. S. and D. E. Lean, "Where is high-lift today? A review of past UK research programmes," in *High-Lift System Aerodynamics*, Neuilly Sur Seine, France: NATO AGARD, 1993. Conference Proceedings 515: Papers presented and discussions recorded at the 71st Fluid Dynamics Panel Meeting and at the Symposium held in Banff, Alberta, Canada, from 5th - 8th October 1992.

[36] Fejtek, I. G., "Computer code validation challenge – multiple element airfoil," in *Proceedings of the Fourth Annual Conference of the CFD Society of Canada*, pp. 733–735, Ottawa, Ontario: CFD Society of Canada, June 2-6 1996.

[37] Fejtek, I., "Summary of code validation results for a multiple element airfoil test case," AIAA Paper 97-1932, June 1997.

[38] Wong, P. and D. W. Zingg, "A Newton-Krylov algorithm for turbulent aerodynamic flows on unstructured grids," Canadian Aeronautics and Space Institute, Apr. 2003. CASI 50th Annual Aerodynamics Conference Symposium, Montreal.

[39] Thibert, J. J., M. Grandjacques and L. H. Ohman, "NACA 0012 airfoil," in *Experimental Data Base for Computer Program Assessment: Report of the Fluid Dynamics Panel Working Group 04* (J. Barche, ed.), chapter A/1 in AGARD Advisory Report #138. Neuilly-Sur-Seine, France: North Atlantic Treaty Organization, May 1979.

[40] Geuzaine, P., I. Lepot, F. Meers and J.-A. Essers, "Multilevel Newton-Krylov algorithms for computing compressible flows on unstructured meshes," AIAA Paper 99-3341, 1999.

[41] van den Berg, B., "Boundary layer measurements on a two-dimensional wing with flap," NLR TR 79009 U, January 1979.

[42] Rizzi, A. and H. Viviand, "Collective comparison of the solutions to the workshop problems," in *Numerical methods for the computation of inviscid transonic flows with shock waves: A GAMM workshop*, Vol. 3 of *Notes on numerical fluid mechanics*, Braunschweig, Wiesbaden: Friedrich Vieweg & Son, 1981.

[43] Various, "Test cases for inviscid flow field methods: Report of fluid dynamics panel working group 07," AGARD Advisory Report #211. Neuilly-Sur-Seine, France: North Atlantic Treaty Organization, May 1985.

[44] Bristeau, M. O. *et al.* , "Introduction," in *Numerical Simulation of Compressible Navier-Stokes Flows: A GAMM Workshop* (M. O. Bristeau, R. Glowinski, J. Periaux and H. Viviand, ed.), Notes on numerical fluid mechanics, Braunschweig, Wiesbaden: Friedrich Vieweg & Sohn Publishing, 1987.

[45] Harris, C. D., "Two-dimensional aerodynamic characteristics of the NACA 0012 airfoil in the Langley 8-foot transonic pressure tunnel," NASA TM 81927, 1981.

[46] Holst, T. L., "Viscous transonic airfoil workshop: Compendium of results," Journal of Aircraft, Vol. 25, No. 12, 1988. See also AIAA Paper 87-1460.

Nomenclature

Δ	The difference operator	$\mathbf{F}$	The inviscid flux tensor
ϵ	A perturbation	$\mathbf{G}$	The stress and heat flux tensor
ϵ_{mz}	Machine zero	γ	Ratio of specific heats

∇	The gradient operator	T_∞	Freestream temperature
$\hat{\mathbf{i}}$	Unit normal aligned with reference chord	τ_{ij}	Stress
$\hat{\mathbf{j}}$	Unit normal perpendicular to reference chord	$_{x,y}$	Subscript denotes partial differentiation
$\mathcal{M}_\infty$	Freestream Mach number	$_{ij}$	Dual subscript denotes relation to ref. chord
μ	Dynamic viscosity	c	Reference chord length
μ_∞	Freestream dynamic viscosity	e	Total internal energy
μ_t	Dynamic turbulent viscosity	i	General case node index subscript
$\hat{\mathbf{n}}$	Generic unit normal vector	k	Outer iteration superscript
ν_t	Kinematic turbulent viscosity	l	Inner iteration superscript
$\bar{\nu}$	Intermediate turbulent viscosity variable	p	Pressure
Ω	A computational domain	R	Universal gas constant
$\partial\Omega$	The boundary of a control volume	T	Temperature
$\mathcal{P}r$	Prandtl number	u	Chord-aligned Cartesian velocity component
$\mathcal{P}r_t$	Turbulent Prandtl number	v	Cartesian velocity normal to chord
$\mathbf{Q}$	A solution vector	x	Coordinate axis, aligned with reference chord
$\mathbf{q}$	Heat flux vector	y	Coordinate axis, normal to reference chord
$\mathcal{R}e$	Reynolds number based on ref. chord length	$\mathbf{v}$	A generic vector
ρ	Density	$\mathbf{R}(\mathbf{Q})$	Residual vector

								Pueyo & Zingg[32]		Present	
Label	Airfoil	$\mathcal{M}_\infty$	α	$\mathcal{R}e$	tr.up	tr.lo	Offwall	Nodes	jm×km	Nodes	foil/farf
PZ1 [42]	NAC	0.63	2.00	—	—	—	$2\cdot10^{-3}$	9,711	249x39	9,657	200/108
PZ2 [43]	NAC	0.80	1.25	—	—	—	$2\cdot10^{-3}$	9,711	”	9,657	”
PZ3 [44]	NAC	0.80	5.00	$5.00\cdot10^{2}$	—	—	$5\cdot10^{-4}$	12,201	249x49	12,303	200/72
PZ4	NAC	0.30	0.00	$2.88\cdot10^{6}$	0.43c	0.43c	$1\cdot10^{-5}$	16,881	331x51	22,371	480/72
PZ5	NAC	0.30	6.00	$2.88\cdot10^{6}$	0.05c	0.80c	$1\cdot10^{-5}$	16,881	”	22,371	”
PZ6 [45, 46]	NAC	0.70	1.49	$9.00\cdot10^{6}$	0.05c	0.05c	$1\cdot10^{-5}$	16,881	”	22,371	”
PZ7	NAC	0.16	12.00	$2.88\cdot10^{6}$	0.01c	0.95c	$1\cdot10^{-5}$	16,881	”	22,371	”
PZ8 [26, 46]	RAE	0.729	2.31	$6.50\cdot10^{6}$	0.03c	0.03c	$1\cdot10^{-5}$	15,279	321x49	13,514	300/72
								Wong & Zingg [38]		Present	
WZ3C [26, 46]	RAE	0.729	2.31	$6.5\cdot10^{6}$	—	—	$2\cdot10^{-6}$	15,900	—	16,948	300/72
								Geuzaine [11]		Present	
G1 [39]	NAC	0.502	1.77	$2.91\cdot10^{6}$	—	—	$3\cdot10^{-5}$	*14,110	—	*16,970	260/44
G3 [41]	NLR	0.185	6.00	$2.51\cdot10^{6}$	—	—	$2\cdot10^{-5}$	*25,661	—	*25,528	380/44
								Chisholm&Zingg[12]		Present	
CZ1	NAC	0.3	6.00	$9.00\cdot10^{6}$	0.05c	0.80c	$1\cdot10^{-6}$	17,385	305x57	17,578	250/92
CZ2 [26, 46]	RAE	0.729	2.31	$6.50\cdot10^{6}$	0.03c	0.03c	$2\cdot10^{-6}$	14,619	257x57	14,685	250/87
CZ3 [41]	NLR	0.185	6.00	$2.51\cdot10^{6}$	—	—	$1\cdot10^{-6}$	44,059	—	41,295	250/200

Table 10. Selected data for external flows. NAC refers to the NACA 0012 airfoil, while RAE refers to the RAE 2822 airfoil. NLR refers to the NLR 7301 modified with flap. Turbulent flow is tripped on the upper surface at *tr.up* and on the lower at *tr.lo*. Two cases (see label G1 and G3) refer to number of cells, instead of nodes; this is indicated by asterisks.