

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, Toronto, Ontario M3H 5T6, Canada

DOI: 10.2514/1.22972

A fast Newton–Krylov algorithm is presented that solves the turbulent Navier–Stokes equations on unstructured 2-D 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 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 1×10^{-12} in 90 s on recent commodity hardware, whereas the lift coefficient is converged to three figures in one quarter of that time.

Nomenclature

c	=	reference chord length
e	=	total internal energy
\mathbf{F}	=	the inviscid flux tensor
\mathbf{G}	=	the stress and heat flux tensor
$\hat{\mathbf{i}}$	=	unit normal aligned with reference chord
$\hat{\mathbf{j}}$	=	unit normal perpendicular to reference chord
M_∞	=	freestream Mach number
$\hat{\mathbf{n}}$	=	generic unit normal vector
Pr	=	Prandtl number
Pr_t	=	turbulent Prandtl number
p	=	pressure
\mathbf{Q}	=	a solution vector
\mathbf{q}	=	heat flux vector
$\mathbf{R}(\mathbf{Q})$	=	residual vector
Re	=	Reynolds number based on reference chord length
T	=	temperature
T_∞	=	freestream temperature
u	=	chord-aligned Cartesian velocity component
v	=	Cartesian velocity normal to chord
\mathbf{v}	=	a generic vector
x	=	coordinate axis, aligned with reference chord
y	=	coordinate axis, normal to reference chord
γ	=	ratio of specific heats
Δ	=	the difference operator
$\partial\Omega$	=	the boundary of a control volume
ϵ	=	a perturbation
ϵ_{mz}	=	machine zero
μ	=	dynamic viscosity
μ_t	=	dynamic turbulent viscosity
μ_∞	=	freestream dynamic viscosity
$\bar{\nu}$	=	intermediate turbulent viscosity variable
ν_t	=	kinematic turbulent viscosity
ρ	=	density
τ_{ij}	=	component of stress tensor
Ω	=	a computational domain

∇ = the gradient operator

Subscripts

i	=	general case node index subscript
ij	=	dual subscript denotes relation to reference chord
x, y	=	subscript denotes partial differentiation

Superscripts

k	=	outer iteration superscript
l	=	inner iteration superscript

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 labeled 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 Venkatakrisnan [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 Venkatakrisnan [2] demonstrated in his seminal work that entry into the region of convergence might be delayed: a transonic Euler case required roughly 40 iterations before rapid convergence. Similar results were shown by Orkwis and McRae [4] and Vanden and Orkwis [5] for hypersonic laminar cases, by Forsyth and Jiang [6] for sub- and supersonic laminar cases, and by Barth and Linton [7] for a high-lift configuration.

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 applied to a Newton-like method for incompressible turbulent flows, as they wrote that quadratic convergence rates are unattainable with such a model. Geuzaine [11]

Presented as Paper 691 at the 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 9–12 January 2006; received 6 February 2006; revision received 1 September 2006; accepted for publication 29 November 2006. Copyright © 2006 by David W. Zingg and Max Blanco. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0001-1452/07 \$10.00 in correspondence with the CCC.

*Doctoral Candidate, 4925 Dufferin Street. Student Member AIAA.

†Corresponding Author, Professor, Senior Canada Research Chair in Computational Aerodynamics, 2004 Guggenheim Fellow, 4925 Dufferin Street. Senior Member AIAA.

implemented the same model in a fully coupled form, as well as a k - ω 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 employed 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 Sec. II, which is then compared with experimental data in Sec. III. We choose and study a metric of comparison in Sec. IV, and discuss the optimization of the algorithm in Sec. 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 Sec. VIII. A table of test cases and relevant data along with labels employed throughout this study completes the paper.

II. Algorithm Description

A. Newton–Krylov 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 italic 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

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

and 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,

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

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

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

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

Once the GMRES, or “inner,” iterations labeled 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 (1–3) 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] Sec. 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 is 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, $\mathbf{Q}_i = [(\rho)_i, (\rho u)_i, (\rho v)_i, (e)_i, (\rho v)_i]^T$. The fundamental difference between the present loosely coupled scheme and fully coupled schemes is the separation of variables. Fully coupled schemes for planar flow have their vectors formed such that the Jacobian block

dimension is five. We write our solution as two distinct processes, both of which employ the same Newton–Krylov algorithm, to solve the mean flow, $\mathbf{1}\mathbf{Q}_i = [(\rho)_i, (\rho u)_i, (\rho v)_i, (e)_i]^T$, independently from the turbulent flow, $\mathbf{2}\mathbf{Q}_i = [(\rho v)_i]^T$. The Jacobian matrix blocks then are of dimension four and one, 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.

B. Navier–Stokes Equations

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

$$\mathbf{R}(\mathbf{Q}) \equiv \oint_{\partial\Omega} (\check{\mathbf{F}} \cdot \hat{\mathbf{n}} \, dS - \check{\mathbf{G}} \cdot \hat{\mathbf{n}} \, dS) = \mathbf{0} \quad (4)$$

where

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

We assume Stokes’ hypothesis holds for air, which behaves in our régime like a Newtonian fluid:

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

where stress terms have been denoted by $\check{\boldsymbol{\tau}}$. Two groups of symbols in Eq. (6) remain to be defined in the next two paragraphs: \mathbf{q} and $\check{\boldsymbol{\tau}}$, with the aid of the Boussinesq and Reynolds analogies. The former analogy is employed to write terms similar to those that involve the dynamic viscosity 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 $\check{\boldsymbol{\tau}}$ can be related to viscosity and the strain rate tensor as

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

in which dynamic viscosity depends on temperature. We follow Schlichting ([20], Fig. 15.1, p. 340) and Mavriplis [21] to write a power law relation to model this feature

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

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

$$\mathbf{q} = -\frac{1}{\gamma - 1} \left[\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t} \right] \nabla T \quad (9)$$

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

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

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, whereas the viscous fluxes are piecewise constant over each triangle, because they are formed from first derivatives of flow variables.

A Riemann-type scheme as detailed by Thomas and Salas [22] is employed at the far-field 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].

C. 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

$$\mu_t \equiv \rho \bar{v} \frac{(\bar{v}/\nu)^3}{(\bar{v}/\nu)^3 + c_{v1}^3} \quad (11)$$

which then allows the variable μ_t to be incorporated into the mean flow as an additional dynamic viscosity and, with division by $Pr_t \equiv 0.90$ [9] (see Schlichting [20], Sec. 19.g), as an additional heat flux term. A BVP is solved for the working variable \bar{v} . 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_{r3} and c_{r4} documented in the article [9], and prefer instead to employ those values listed in [8]. Values of \bar{v} in the freestream on the far-field boundary are set to $v_\infty = 0.01$, 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 1) each system makes appeal to different physical arguments, 2) each system has its own numerical properties, and 3) investment is reduced in case a change in model is desired.

D. Grid Sequencing

Most researchers who have reported Newton-like algorithms for flow solution employ grid sequencing (first demonstrated 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 which have been employed with success elsewhere. The grids that were employed were created independently in a preprocessing step. We employ a sequence of grids in which the coarsest grid contains approximately 1×10^3 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.

E. 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 startup strategies reported in the literature, usually some sort of pseudotime 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.

A. Test Cases and Grids

The airfoils are represented by coordinate pairs, which we place on natural cubic splines. The external far-field 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 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.

B. Comparison to Experiment

We repeat with our algorithm two experimental tests, summary data of which can be found in Table 1. We compare our algorithmic data 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 No. 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 Fig. 1a. Case 6 of Cook et al. [26] had nominal Mach number 0.725, angle of attack 2.92 deg. We calculate the flow with Mach number 0.729, angle of attack 2.31 deg, 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 Fig. 1b. The grid was composed 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

Table 1 Parameters for comparison to experiment of Fig. 1

Label	Case	M_∞	α , deg	Re	Offwall	No. of nodes	No. on foil/farfield
WZ3C [26]	Airfoil RAE	0.729	2.31	$6.50 \cdot 10^6$	$2 \cdot 10^{-6}$	16,948	300/72
LIT2 [27]	NLHP 3elem	0.197	4.01	$3.52 \cdot 10^6$	$2.5 \cdot 10^{-6}$	51,779	1376/82

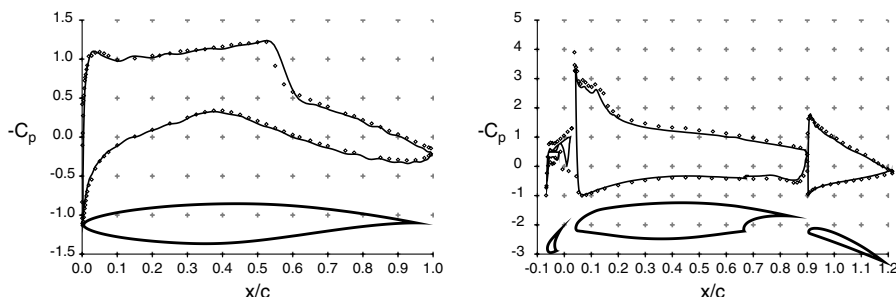


Fig. 1 Calculated surface coefficients of pressure C_p compared with overlays of those measured by a) Cook et al. [26] and b) Moir [27].

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. $R(Q)$ ” in tabular form, should transcend machinery and makes algorithm comparison somewhat possible. Pueyo and Zingg [32] remark of one of its weaknesses: “. . . it tends to favor 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 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 digits in roughly one-quarter of the time to reduce the residual to 10^{-12} .

The present algorithm 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 $R(Q)$ score than the two AMD machines tested. All subsequent data are reported for an AMD Athlon XP, which clocked 2167 MHz.

Table 2 Machinery used for test of present algorithm

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

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

Label	Eq. $R(Q)$	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

V. Optimization of Loosely Coupled Algorithm

A. Turbulence Model ILU(n)

Geuzaine [11] reports 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 Sec. III.A.

For each mean-flow outer iteration, the mean-flow residual is reduced by a prescribed tolerance, usually 0.1. We then perform a number of turbulence model outer iterations such that the turbulent residual is less than the mean-flow residual. The fully coupled algorithms with which we contrast here have no such freedom. The residuals are defined and tracked separately for the two processes. The turbulence model residual, $\|_2 R^k\|$, is checked after every turbulence model outer iteration to see whether it is lower than the mean-flow residual, $\|_1 R^k\|$. The turbulence model outer iteration is forced to repeat if this criterion is not met, to a maximum of five repeated outer iterations. We limit the number of turbulence model outer iterations to five, although cases in which the limit is reached are rare. This is illustrated for case PZ8 in Fig. 2.

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), whereas multi-element cases may require one or two more levels of fill.

Table 4 and Fig. 3 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 of 500 inner iterations to converge to a residual of 1×10^{-12} . Net inner iterations for the turbulence model are shown in the table as Σ_{in} . Wallclock time is minimized for $n = 5$ or $n = 7$. However, for this problem we are more interested in a tradeoff

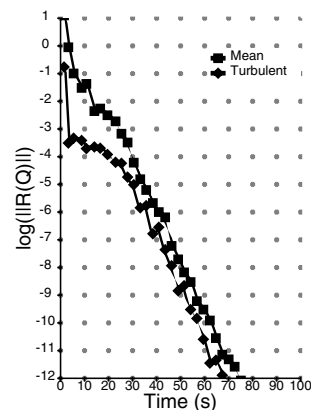


Fig. 2 Convergence history for Case PZ8. Mean flow and turbulence model residuals of optimal solution.

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

ILU(n)	MF	TM	Σ_{in}	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

between storage requirements and performance, and thus we choose as turbulence model preconditioners a range between ILU(2) and ILU(4).

B. Grid Sequencing

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

The benefit is quite small for single-element cases (Table 6). In fact, for case CZ2 the cost of computing the solution on the coarsest grid exceeds its benefit. The benefit is much greater for the multi-element airfoil (Table 7), perhaps due to the more complex flowfield.

VI. Calculation of a Multi-Element Airfoil Flow

A takeoff geometry from the U.K. National High-Lift Programme is depicted in Fig. 1b. The wind tunnel measurement data were published in 1994 by the AGARD as chapter A/2 of Advisory Report No. 303 [27]. The slat, denoted “L1” by the experimenters [35], and flap, denoted “T2,” are examined for the case in which the airfoil was set at an angle of attack of 4.01 deg to the oncoming flow. Fejtek reported a group of results [36,37] for this case. The freestream Mach number was $M_\infty = 0.197$, whereas the Reynolds number calculated to $Re = 2.51 \times 10^6$ based on the nested configuration chord length c . The leading-edge slat, positioned at an angle of 25 deg, measured $12.5\%c$, whereas the flap measured $33\%c$ and was positioned with a deflection angle of 20 deg.

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 2, whereas storage requirements only increased by 25% when we increased the fill level

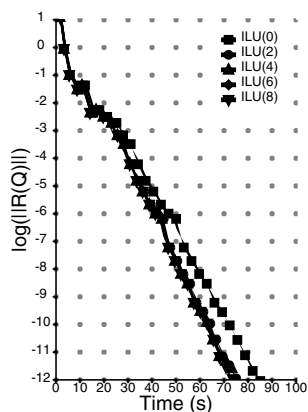


Fig. 3 Convergence histories for Case PZ8. Effect of turbulence model ILU fill on overall solution time. The turbulence model residual has been omitted for clarity.

Table 5 Residual at which to sequence for case PZ8

Sequence at	Time, s
1×10^{-1}	76
1×10^{-2}	63
1×10^{-3}	68
1×10^{-4}	69
1×10^{-5}	70
1×10^{-6}	71
1×10^{-7}	72
1×10^{-8}	72
1×10^{-9}	73
1×10^{-10}	74

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 1×10^{-12}

Case	No. grids	Time, s	$R(Q)$
PZ5	1	121.8	1220
PZ5	2	112.1	1130
PZ5	3	110.2	1100
CZ2	1	91.6	1150
CZ2	2	83.8	1050
CZ2	3	89.9	1130

Table 7 Case LIT2: Equivalent residual evaluations and other statistics for 10 order of magnitude residual reduction

Case	No. grids	Time, s	$R(Q)$	GMRES	BILU(n)	Storage, Mb
LIT2	1	1340	4970	(50,0)	4	328
LIT2	3	530	2048	(50,0)	6	388

to BILU(6) and employed a sequence of three grids for this complex flow. The surface coefficient of pressure is depicted in Fig. 1b. We can solve this flow with a Pentium class computer in under ten minutes.

VII. Comparison with Fully Coupled Schemes

A. 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 2-D grid creation algorithm. An icosahedral extrapolation for 3-D 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.

B. Equivalent $R(Q)$ Comparison

We compare in Table 9 our scheme to various others that have appeared in the literature. Data are included for Geuzaine [11], Wong and Zingg [43] 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.

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 fifth column of Table 9, and in each case we employed the same number or less. The CPU time for L_2 -norm convergence to 1×10^{-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. [44], labeled 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 [46], contains 14,110 cells, and requires

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

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

Table 9 Equivalent residual evaluations $R(Q)$ for four Newton–Krylov solvers are listed in the final four columns of the table; number of grid nodes, offwall spacing, and convergence criteria are consistent for each case; references and further grid information may be found in Table 10

Label	Foil	M_∞	α	Grids	Equivalent $R(Q)$			
					Present	Chisholm and Zingg [12]	Geuzaine [11]	Wong and Zingg [43]
G1	NAC	0.502	1.77	3	630	—	1,500	—
CZ1	NAC	0.300	6.00	3	900	900	—	—
WZ3C	RAE	0.729	2.31	1	1,300	—	—	3,000
CZ2	RAE	0.729	2.31	3	1,120	900	—	—
G3	NLR	0.185	6.00	1	1,220	—	10,000	—
CZ3	NLR	0.185	6.00	3	1,450	1,500	—	—

Table 10 Selected data for external flows. NAC refers to the NACA 0012 airfoil, whereas 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.

Label	Airfoil	M_∞	α	Re	$tr.up$	$tr.lo$	Offwall	Nodes		Nodes	foil/farf
								jm \times km	Present		
								Pueyo and Zingg [32]			
PZ1 [38]	NAC	0.63	2.00	—	—	—	2×10^{-3}	9,711	249 \times 39	9,657	200/108
PZ2 [39]	NAC	0.80	1.25	—	—	—	2×10^{-3}	9,711	249 \times 39	9,657	200/108
PZ3 [40]	NAC	0.80	5.00	5.00×10^2	—	—	5×10^{-4}	12,201	249 \times 49	12,303	200/72
PZ4	NAC	0.30	0.00	2.88×10^6	0.43c	0.43c	1×10^{-5}	16,881	331 \times 51	22,371	480/72
PZ5	NAC	0.30	6.00	2.88×10^6	0.05c	0.80c	1×10^{-5}	16,881	331 \times 51	22,371	480/72
PZ6 [41,42]	NAC	0.70	1.49	9.00×10^6	0.05c	0.05c	1×10^{-5}	16,881	331 \times 51	22,371	480/72
PZ7	NAC	0.16	12.00	2.88×10^6	0.01c	0.95c	1×10^{-5}	16,881	331 \times 51	22,371	480/72
PZ8 [26,42]	RAE	0.729	2.31	6.50×10^6	0.03c	0.03c	1×10^{-5}	15,279	321 \times 49	13,514	300/72
								Wong and Zingg [43]			
WZ3C [26,42]	RAE	0.729	2	6.5×10^6	—	—	2×10^{-6}	15,900	—	16,948	300/72
								Geuzaine [11]			
G1 [44]	NAC	0.502	1.77	2.91×10^6	—	—	3×10^{-5}	*14,110	—	*16,970	260/44
G3 [45]	NLR	0.185	6.00	2.51×10^6	—	—	2×10^{-5}	*25,661	—	*25,528	380/44
								Chisholm and Zingg [12]			
CZ1	NAC	0.3	6.00	9.00×10^6	0.05c	0.80c	1×10^{-6}	17,385	305 \times 57	17,578	250/92
CZ2 [26,42]	RAE	0.729	2.31	6.50×10^6	0.03c	0.03c	2×10^{-6}	14,619	257 \times 57	14,685	250/87
CZ3	NLR	0.185	6.00	2.51×10^6	—	—	1×10^{-6}	44,059	—	41,295	250/200

2.5 times as many function evaluations as our loosely coupled algorithm. For this case, the present algorithm converged to 10^{-12} in 25 s.

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 s.

Case WZ3C: Wong and Zingg [43] report an algorithm designed to solve on grids composed 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 s.

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 Sec. V. The time to convergence of our loosely coupled algorithm is 91 s.

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 [45]. The grid employed by Geuzaine has an offwall spacing of

20 microchords [46]. He converges his turbulence model three orders of magnitude higher than we do. His density residual converges to 1×10^{-10} for this case. We converge the mean-flow residual to 1×10^{-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 s.

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 s.

VIII. 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 (Sec. VII.B).

2) The present unstructured loosely coupled algorithm requires *a similar number of equivalent function evaluations* to a structured fully coupled algorithm (Sec. VII.B).

3) The present loosely coupled algorithm requires *30% less memory* than a fully coupled algorithm (Sec. VII.A).

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

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., Hafez, M., and Ahmad, J., "Calculations of Viscous Flow with Separation Using Newton's Method and Direct Solver," *AIAA Journal*, Vol. 28, No. 5, May 1990, pp. 937–939.
- [2] Venkatakrishnan, V., "Newton Solution of Inviscid and Viscous Problems," *AIAA Journal*, Vol. 27, No. 7, July 1989, pp. 885–891; see also AIAA Paper 88-0413.
- [3] Saad, Y., and Schultz, M. H., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869.
- [4] Orkwis, P. D., and McRae, D. S., "Newton's Method Solver for High-Speed Viscous Separated Flow Fields," *AIAA Journal*, Vol. 30, No. 1, Jan. 1992, pp. 78–85.
- [5] Vanden, K. J., and Orkwis, P. D., "A Comparison of Numerical and Analytical Jacobians," *AIAA Journal*, Vol. 34, No. 6, June 1996, pp. 1125–1130.
- [6] Forsyth, P. A., and Jiang, H., "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 Linton, S. W., "An Unstructured Mesh Newton Solver for Compressible Fluid Flow and its Parallel Implementation," AIAA Paper 95-0221, Jan. 1995.
- [8] Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," AIAA Paper 92-0439, 1992.
- [9] Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aéropatiale*, No. 1, 1994, pp. 5–21.
- [10] Anderson, W. K., Rausch, R. D., and Bonhaus, D. L., "Implicit/Multigrid Algorithms for Incompressible Turbulent Flows on Unstructured Grids," *Journal of Computational Physics*, Vol. 128, No. 2, Oct. 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 Zingg, D. W., "Start-Up Issues in a Newton-Krylov Algorithm for Turbulent Aerodynamic Flows," AIAA Paper 2003-3708, June 2003.
- [13] Smith, T., Hooper, R., Ober, C., Lorber, A., and Shadid, J., "Comparison of Operators for Newton-Krylov Method for Solving Compressible Flows on Unstructured Meshes," AIAA Paper 2004-0743, Jan. 2004.
- [14] Blanco, M., and Zingg, D. W., "Fast Newton-Krylov Method for Unstructured Grids," *AIAA Journal*, Vol. 36, No. 4, 1998, pp. 607–612.
- [15] Cuthill, E., and McKee, J., "Reducing the Bandwidth of Sparse Symmetric Matrices," *Proceedings of the 24th National Conference ACM*, ACM Press, New York, 1969, pp. 157–172.
- [16] Liu, W.-H., and Sherman, A. H., "Comparative Analysis of the Cuthill-McKee and the Reverse Cuthill-McKee Ordering Algorithms for Sparse Matrices," *SIAM Journal on Numerical Analysis*, Vol. 13, No. 2, April 1976, pp. 198–213.
- [17] Gibbs, N. E., Poole, W. G., Jr., and Stockmeyer, P. R., "An Algorithm for Reducing the Bandwidth and Profile of a Sparse Matrix," *SIAM Journal on Numerical Analysis*, Vol. 13, No. 236, 1976, pp. 236–249.
- [18] Balay, S., Gropp, W., McInnes, L. C., and Smith, B., "PETSc 2.0 Users' Manual," Argonne National Laboratory ANL-95/11, Argonne, IL, 1998. Revision 2.0.22.
- [19] Meijerink, J. A., and van der Vorst, H. A., "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, pp. 148–162.
- [20] Schlichting, H., *Boundary Layer Theory*, 4th ed., McGraw-Hill Book Company, New York, 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 Salas, M. D., "Far-Field Boundary Conditions for Transonic Lifting Solutions to the Euler Equations," *AIAA Journal*, Vol. 24, July 1986, pp. 1074–1080.
- [23] Hirsch, C., *Numerical Computation of Internal and External Flows*, Vol. 2, John Wiley & Sons, New York, 1990.
- [24] Barche, J., "Experimental Data Base for Computer Program Assessment: Report of the Fluid Dynamics Panel Working Group 04," AGARD Advisory Report No. 138, NATO, Neuilly-Sur-Seine, France, May 1979.
- [25] Burt, M. (ed.), "A Selection of Experimental Test Cases for the Validation of CFD Codes," AGARD Advisory Report No. 303, NATO, Neuilly-Sur-Seine, France, Aug. 1994.
- [26] Cook, P. H., McDonald, M. A., and Firmin, M. C. P., "Airfoil RAE 2822—Pressure Distributions, and Boundary Layer Wake Measurement," *Experimental Data Base for Computer Program Assessment: Report of the Fluid Dynamics Panel Working Group 04*, edited by J. Barche, NATO, Neuilly-Sur-Seine, France, May 1979, Chap. A/6.
- [27] Moir, I. R. M., "Measurements on a Two-Dimensional Aerofoil with High-Lift Devices," *A Selection of Experimental Test Cases for the Validation of CFD Codes*, edited by M. Burt, NATO, Neuilly-Sur-Seine, France, Aug. 1994, Chap. A/2.
- [28] Walsh, P. C., and Zingg, D. W., "On the Accuracy of Viscous Airfoil Computations Using Solution-Adaptive Unstructured Grids," AIAA Paper 97-0329, Jan. 1997.
- [29] Barth, T. J., "Aspects of Unstructured Grids and Finite-Volume Solvers for the Euler and Navier-Stokes Equations," AGARD Report No. 787: Special Course on Unstructured Grid Methods for Advection Dominated Flows, NATO, Neuilly-sur-Seine, France, May 1992, Chap. 6.
- [30] Barth, T. J., "Steiner Triangulation for Isotropic and Stretched Elements," AIAA Paper 95-0213, 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, Feb. 1988, pp. 181–197.
- [32] Pueyo, A., and Zingg, D. W., "An efficient Newton-GMRES solver for aerodynamic computations," AIAA Paper 97-1955, July 1997.
- [33] Chisholm, T. T., and Zingg, D. W., "A fully coupled Newton-Krylov solver for turbulent aerodynamic flows," in *Proceedings of the 2002 Congress of the International Council of Aeronautical Sciences*, 2002, pp. 333.1–333.9.
- [34] Pueyo, A., and Zingg, D. W., "Efficient Newton-Krylov Solver for Aerodynamic Computations," *AIAA Journal*, Vol. 36, No. 11, 1998, pp. 1991–1997.
- [35] Woodward, D. S., and Lean, D. E., "Where is High-Lift Today? A Review of Past UK Research Programmes," *High-Lift System Aerodynamics*, AGARD CP No. 515, NATO, Neuilly-Sur-Seine, France, 1993, Chap. 1.
- [36] Fejtek, I. G., "Computer Code Validation Challenge—Multiple Element Airfoil," *Proceedings of the Fourth Annual Conference of the CFD Society of Canada*, CFD Society of Canada, Ottawa, Ontario, June 1996, pp. 733–735.
- [37] Fejtek, I. G., "Summary of Code Validation Results for a Multiple Element Airfoil Test Case," AIAA Paper 97-1932, June 1997.
- [38] Rizzi, A., and Viviand, H., "Collective Comparison of the Solutions to the Workshop Problems," *Numerical Methods for the Computation of Inviscid Transonic Flows with Shock Waves: A GAMM Workshop*, Vol. 3, Notes on Numerical Fluid Mechanics, Friedrich Vieweg & Son, Braunschweig, Wiesbaden, 1981.
- [39] Yoshihara, H., and Sacher, P., "Test Cases for Inviscid Flow Field Methods: Report of Fluid Dynamics Panel Working Group 07," AGARD Advisory Report No. 211, NATO, Neuilly-Sur-Seine, France, May 1985.

- [40] Bristeau, M. O., Glowinski, R., Périaux, J., and Viviand, H., "Introduction," *Numerical Simulation of Compressible Navier-Stokes Flows: A GAMM Workshop*, edited by M. O. Bristeau, R. Glowinski, J. Périaux, and H. Viviand, Vol. 18, Notes on Numerical Fluid Mechanics, Friedrich Vieweg & Son, Braunschweig, Wiesbaden, 1987, chap. 1.
- [41] Harris, C. D., "Two-Dimensional Aerodynamic Characteristics of the NACA 0012 Airfoil in the Langley 8-Foot Transonic Pressure Tunnel," NASA TM 81927, 1981.
- [42] Holst, T. L., "Viscous Transonic Airfoil Workshop: Compendium of Results," *Journal of Aircraft*, Vol. 25, No. 12, 1988, pp. 1073–1087; see also AIAA Paper 87-1460.
- [43] Wong, P., and Zingg, D. W., "A Newton-Krylov Algorithm for Turbulent Aerodynamic Flows on Unstructured Grids," *CASI 50th Annual Aerodynamics Conference Symposium, Montreal, Canadian Aeronautics and Space Institute, Ottawa, Ontario, April 2003*.
- [44] Thibert, J. J., Grandjacques, M., and Ohman, L. H., "NACA 0012 Airfoil," *Experimental Data Base for Computer Program Assessment: Report of the Fluid Dynamics Panel Working Group 04*, edited by J. Barche, NATO, Neuilly-Sur-Seine, France, May 1979, Chap. A/1.
- [45] van den Berg, B., "Boundary Layer Measurements on a Two-Dimensional Wing with Flap," National Aerospace Laboratory TR 79009 U, Amsterdam, Jan. 1979.
- [46] Geuzaine, P., Lepot, I., Meers, F., and Essers, J.-A., "Multilevel Newton-Krylov Algorithms for Computing Compressible Flows on Unstructured Meshes," AIAA Paper 99-3341, June 1999.

X. Zhong
Associate Editor