Computers & Fluids 100 (2014) 278–290

Contents lists available at ScienceDirect

Computers & Fluids

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

Self-Adaptive Newton-based iteration strategy for the LES of turbulent multi-scale flows

F. Daude [a,1], I. Mary [a,*], P. Comte [a,b]

[a] Office National d'Études et de Recherches Aérospatiale (ONERA), 29, Avenue de la Division Leclerc, 92322 Châtillon Cedex, France
[b] Institut P', CNRS UPR 3346, ENSMA, Université de Poitiers, 43, route de l'Aérodrome, F-86036 Poitiers, France

ARTICLE INFO

Article history:
Received 26 February 2013
Received in revised form 29 January 2014
Accepted 25 April 2014
Available online 21 May 2014

This work is dedicated to the memory of Prof. Pierre Comte (1961–2011).

Keywords:
Implicit time integration
Approximate Newton method
Self-adaptive iterative process
Large-eddy simulation
Compressible flows
Multi-scale problems

ABSTRACT

An improvement of the efficiency of implicit schemes based on Newton-like methods for the simulation of turbulent flows by compressible LES or DNS is proposed. It hinges on a zonal Self-Adaptive Newton method (hereafter denoted SAN), capable of taking advantage of Newton convergence rate heterogeneities in multi-scale flow configurations due to a strong spatial variation of the mesh resolution, such as transitional or turbulent flows controlled by small actuators or passive devices. Thanks to a predictor of the local Newton convergence rate, SAN provides computational savings by allocating resources in regions where they are most needed. The consistency with explicit time integration and the efficiency of the method are checked in three test cases:

- The standard test-case of 2-D linear advection of a vortex, on three different two-block grids.
- Transition to 3-D turbulence on the lee-side of an airfoil at high angle of attack, which features a challenging laminar separation bubble with a turbulent reattachment.
- A passively-controlled turbulent transonic cavity flow, for which the CPU time is reduced by a factor of 10 with respect to the baseline algorithm, illustrates the interest of the proposed algorithm.

1. Introduction

Large-eddy (LES) and direct simulations (DNS) have been successfully used over the last decade to study academic turbulent flows. With the improvements of both numerical methods and supercomputer capacities, these models are now applied to more complex flow cases, with some industrial relevance, in order to provide reliable results which cannot be given by experiments. The next generation of flow problems tackled by DNS/LES involves multi-scale problems, such as flow control cases, implying the simulation of multiple temporal and spatial scales, which is very challenging for computational physics. Development in efficient numerical algorithms adapted to such problems is of interest for both researchers and engineers.

The time-explicit schemes used traditionally for simulation of compressible flows are limited to time steps that satisfy a stability requirement based on the following generalized CFL number taking to account both convection and diffusion (molecular and turbulent parts in the momentum and energy equations, the latter yields more stringent time step limitations): $\mathrm{CFL} = \max\left(\mathcal{C}^{|u|+c}, \mathcal{C}^{|u|}, \mathcal{D}^{\nu}\right)$ where in 1-D

$$\mathcal{C}^{|u|+c} = \frac{|u|+c}{\Delta x}\Delta t, \quad \mathcal{C}^{|u|} = \frac{|u|}{\Delta x}\Delta t \quad \text{and} \quad \mathcal{D}^{\nu} = 2\gamma\frac{\mu}{\rho \mathrm{Pr}\Delta x^2}\Delta t$$

with the Courant numbers linked to the acoustic and convective velocities ($\mathcal{C}^{|u|+c}$ and $\mathcal{C}^{|u|}$, respectively) and the non-dimensional dissipation, generically denoted $\mathcal{D}^{\nu}$ with a possibly mixed definition of μ/Pr depending on the turbulence modeling approach used. Due to high local mesh refinements needed to capture the motion of turbulent scales close to walls and around small geometries like control devices, this stability requirement leads to very small time steps, in general because of $\mathcal{C}^{|u|+c}$, making explicit schemes less efficient than in simpler flow configurations and geometries (homogeneous or free-shear turbulence, uncontrolled wall-bounded flows, etc.). Implicit time integration, of widespread usage in (U)RANS type CFD, is increasingly appealing in this context. As shown in [1,2], caution is needed in the choice of the time step in the DNS/LES context, since too large values can yield excessive dissipation, even with non-dissipative spatial discretization. However, the suitability of implicit schemes has been demonstrated in DNS of wall-bounded turbulence

* Corresponding author. Tel.: +33 1 46 73 42 69; fax: +33 1 46 73 41 66.
E-mail address: ivan.mary@onera.fr (I. Mary).
[1] Present address: LaMSID – UMR EDF/CNRS/CEA 8193, 1 Av. du Général De Gaulle, 92141 Clamart, France.

http://dx.doi.org/10.1016/j.compfluid.2014.04.028
0045-7930/

[3–5] and in LES of complex flows [6–11]. In these cases, the quality of the results suggests that the dissipation brought about by large CFL values (w.r.t. those permitted by explicit time integration with the same discretizations in space) predominantly affects phenomena that are not crucial to the dynamics investigated (e.g. sound waves in a non-resonant low-speed aerodynamic problem).

Most implicit simulations use a second-order time accurate scheme and a Newton-like method [3,5–8,11] with incomplete convergence (i.e. with convergence residuals significantly larger than machine zero), the influence of which has to be investigated. Below a threshold value of the residual, which is *a priori* problem dependent, incomplete convergence of the Newton process is expected not to be significantly more problematic than round-off errors in 64-bit floating point machines, that is, have little influence on the flow solution, at least in a statistical sense. It has been noticed in several studies [12,13] that the CFL number strongly affects the convergence speed of the Newton process. For multi-scale problems, heterogeneity of cell sizes yields disparate local CFL values, detrimental to the conditioning of the linear system to be solved in the inner iterative process. With standard Newton methods, the number of inner iterations is determined by the highest local CFL value, and is applied to all points (cells or elements) of the domain, hence a certain waste of numerical resources in low CFL regions.

The Self-Adaptive Newton method proposed hereafter (and denoted SAN) aims at gaining efficiency by means of a zonal determination of the number of iterations, with the same overall value of the convergence residual as the baseline Newton method. This algorithm is presented in details in Section 2 for a general set of partial derivative equations. Choices specific to compressible LES are briefly discussed in Section 3, in which the influence of the incomplete Newton convergence is also analyzed in the case of a simplified 1-D advection equation. In Section 4, a detailed validation of the SAN method is performed in two dimensions, in the case of the linear advection of isentropic vortex on heterogeneous grids.

Two realistic applications are then considered. The first one (Section 5) is a transitional/turbulent boundary layer on the suction side of an airfoil. The configuration selected is known to be particularly sensitive to small perturbations, and comparison of the turbulent statistics obtained with counterparts obtained with high-order explicit time integration and the same discretizations in space is expected to make a convincing proof of the validity of the method. This problem is however not multi-scale enough to highlight the gain in efficiency.

We thus propose a simple configuration of controlled flow, namely a high-subsonic flow over a deep cavity passively controlled by means of a small spanwise cylinder (Section 6), for which high-quality experimental data are available [14], and for which SAN cuts the CPU time by a factor of about 10.

2. Newton method

2.1. Iterative process

For an unsteady physical problem modeled by a nonlinear set of PDEs, the use of an implicit time integration method leads to a nonlinear fixed-point problem at each time step of the form:

$$\mathcal{F}\left(\boldsymbol{U}^{n+1}\right) = \boldsymbol{0} \tag{1}$$

where $\mathcal{F}$ represents both the spatial and temporal operators, n is the time level and $\boldsymbol{U}$ is the state vector to be found. This nonlinear problem is usually solved via a Newton iterative procedure:

$$\begin{cases} \mathcal{J}^{(l)}\Delta\boldsymbol{U}^{(l)} &= -\mathcal{F}\left(\boldsymbol{U}^{(l)}\right) = \boldsymbol{\mathcal{R}}^{(l)} \\ \boldsymbol{U}^{(l+1)} &= \boldsymbol{U}^{(l)} + \Delta\boldsymbol{U}^{(l)} \end{cases} \qquad l = 0, 1, \ldots \tag{2}$$

where $\mathcal{J}^{(l)} = \partial_{\boldsymbol{U}}\mathcal{F}\left(\boldsymbol{U}^{(l)}\right)$ is the Jacobian, $\boldsymbol{\mathcal{R}}^{(l)} \in \mathrm{IR}^p$ is called the Newton residual, and p represents the degrees of freedom at each time step. For unsteady problems, the initialization of this iterative process is classically done via $\boldsymbol{U}^{(l=0)} = \boldsymbol{U}^n$. For highly coupled problems, the computation and the inversion of the exact Jacobian matrix, $\mathcal{J}$, are often very expensive in term of CPU time. Therefore practical Newton methods are usually based on a Jacobian-free approach [15] or on an approximate iterative Newton–Raphson method as in [3,5,6,11]. Convergence to machine accuracy at each time step may be time-consuming, and not required for the accuracy of the solution. So, in practice, the nonlinear problem (1) is solved within a small tolerance. For example in the context of LES or DNS, Rai and Moin [3] with the second-order time accurate backward differentiation (Gear) scheme apply between two and four inner iterations per time step to ensure the convergence of the Newton method. In the same way, Rizzetta et al. [11] use three Newton iterations with the same time integration. Weber and Ducros [6] with the second-order Crank–Nicolson scheme stop the iterative process when the L_2 norm of the Newton residual on density is reduced by three orders of magnitude. Martín and Candler [5] with the Gear method use a convergence criterion based upon the per-point average L_1 norm of the density convergence error with a practical limit of 20 iterations of the Newton process. The following convergence criterion based on a drop in the L_∞ norm of the Newton residual is retained in this study:

$$\frac{\left|\left|\boldsymbol{\mathcal{R}}^{(N^{n+1}-1)}\right|\right|_\infty}{\left|\left|\boldsymbol{\mathcal{R}}^{(0)}\right|\right|_\infty} \leqslant \varepsilon \tag{3}$$

where ε is the user-specified convergence tolerance which is problem dependent [5] and N^{n+1} is the inner number of iterations when this tolerance is achieved (and the iterative process stops). Finally, the update of the numerical solution is given via $\boldsymbol{U}^{n+1} = \boldsymbol{U}^{(N^{n+1})}$.

2.2. Self-Adaptive Newton (SAN) method

2.2.1. Motivation

The computational effort to solve the nonlinear unsteady problem with a targeted convergence criterion is generally driven by the highest local CFL value: on the one hand the number of nonlinear iterations increases with CFL because Jacobian approximations are less accurate; on the other hand, the resolution of the linear system becomes more costly. However, for multi-scale problems characterized by a high ratio between the smallest and the largest cell of the whole domain, only a few cells have a high value of the CFL number. So, during the Newton process, the local residual is as spatially heterogeneous as the local CFL value, leading to a kind of "oversolving" problem [16]. Thus, a Self-Adaptive Newton (SAN) method is proposed to appreciably reduce the cost of simulations by limiting the oversolving problem especially when the grid used for the computation presents very disparate cells size. The goal of SAN is to equidistribute the convergence errors by allocating resources in regions where they are most needed.

Originally, multi-block solvers were developed to handle complex geometries with structured blocks. This characteristic is also used to adapt the turbulence modeling, thanks to zonal RANS/LES coupling or to perform parallel computing. More recently, Tòth et al. [17] proposed to couple explicit and implicit time integration methods in order to adapt the algorithm to the CFL number of each block. The idea of the proposed SAN strategy relies on adapting the frequency of the Newton iteration computation in each of the blocks of the grid in function of their local convergence rate. In the following, the whole computational domain Ω is considered to be subdivided into non-overlapping subdomains denoted Ω_d for $d = 1, \ldots, D : \Omega = \bigcup_{d=1}^{D}\Omega_d$.