Computational Fluid Dynamics JOURNAL vol.10 No.4 January 2002 (pp.548–552)
Selected Papers from Asian CFD Conference 4 and 3rd China-Japan CFD Workshop, Mianyang September 2001

MONOTONE INTEGRATED LARGE EDDY SIMULATION FOR SUPERSONIC BOUNDARY LAYER FLOWS

N QIN† Y H CHONG†

Abstract

Monotone integrated large eddy simulation (MILES) has been carried out to simulate supersonic turbulent boundary layer flows for relatively high Reynolds numbers. Two Riemann solver based methods, Osher's and Roe's schemes, were tested along with a MUSCL scheme to investigate their suitability for MILES simulation. The effects of different limiters used in the MUSCL scheme were also tested regarding the MILES simulations.

1 INTRODUCTION

For turbulent flow simulation, if the grid space and the time step are not sufficiently small to resolve all the spatial and temporal development of turbulence, some mechanism needs to be introduced to model the physical dissipation of the grid scale eddies. Conventional large eddy simulation(LES) uses subgrid-scale modelling for the physical dissipation. For every dynamical variable u (e.g. pressure), one explicitly defines a smoothed or 'filtered' variable $\overline{u}$ defined as an appropriate local average of u, so that $\overline{u}$ is a smoother function that only follows the large scale structure of u and lacks the small scale fluctuations. One then derives from the Navier-Stokes equations a set of equations satisfied by the corresponding filtered variables. These equations are *exact* as no approximation has been made so far. However, the LES equations are unclosed. At this stage an approximate model is chosen for the unclosed terms to give a set of model LES equations. These equations are then solved numerically.

Another approach to LES developed recently is to use the inherent dissipation of numerical schemes to mimic transfer of energy to smaller scales. Monotone integrated large eddy simulation (MILES) takes such approach. Even though there have been other publications which can be classified as MILES (Kawamura and Kuwahara [10] in incompressible case and Porter *et al.* [1] in compressible case), Boris *et al.* [6, 2] is generally credited for introducing the MILES concept. Boris *et al.* used the Flux-Corrected Transport (FCT) scheme in a cylinderical jet entrainment simulation to show that some high-resolution monotone algorithms have intrinsic subgrid turbulence models built into it, which plays the role of transferring the energy to subgrid scales.

Garnier *et al.* [3] tested several shock-capturing schemes (Jameson, TVD-MUSCL, ENO) to evaluate the relevance of the use of such schemes for Euler solutions in the LES context for freely decaying isotropic turbulence simulation. Although some known physical trends were respected, it is found that the small scales of the simulated flow suffer from numerical damping. They also compared the numerical dissipation and the dissipation provided by the SGS models (Smagorinsky or dynamic SGS models), and concluded that addition of explicit subgrid-scale (SGS) models to shock-capturing schemes is unnecessary for the cases tested since it is in fact overshadowed by the intrinsic numerical dissipation.

A solid wall boundary posses a great challenge for LES. In this case, dynamic SGS modelling [8] shows better results than the Smagorinsky SGS model. Some new models [13] were proposed but it is still problematic for wall-bounded flows. This paper investigates the behaviour of two Riemann solver based methods along with the MUSCL approach for MILES simulation for supersonic boundary layers.

Received on November 30, 2000.
† Centre for Computational Aerodynamics, Cranfield University College of Aeronautics; Bedford MK43 0AL, England
email: n.qin@cranfield.ac.uk

Methodology

Numerical Methods

Basic difference between conventional LES and MILES is filtering. In the conventional LES, cell values are obtained from successive filtering and cell averaging:

$$\overline{u}_i(\mathbf{x}) = \iiint G\left(\mathbf{x}-\mathbf{x}'\right) u_i\left(\mathbf{x}'\right) d\mathbf{x}' \qquad (1)$$