Journal of Computational Physics 316 (2016) 652–681

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A high-order multi-zone cut-stencil method for numerical simulations of high-speed flows over complex geometries

Patrick T. Greene *, Jeff D. Eldredge, Xiaolin Zhong, John Kim

Department of Mechanical and Aerospace Engineering, University of California, Los Angeles, 420 Westwood Plaza, Los Angeles, CA, 90095, USA

A R T I C L E   I N F O

Article history:
Received 2 June 2014
Received in revised form 3 February 2016
Accepted 16 April 2016
Available online 20 April 2016

Keywords:
Cut-stencil
Multi-zone refinement
High-order
Complex geometry
Cartesian grid

A B S T R A C T

In this paper, we present a method for performing uniformly high-order direct numerical simulations of high-speed flows over arbitrary geometries. The method was developed with the goal of simulating and studying the effects of complex isolated roughness elements on the stability of hypersonic boundary layers. The simulations are carried out on Cartesian grids with the geometries imposed by a third-order cut-stencil method. A fifth-order hybrid weighted essentially non-oscillatory scheme was implemented to capture any steep gradients in the flow created by the geometries and a third-order Runge–Kutta method is used for time advancement. A multi-zone refinement method was also utilized to provide extra resolution at locations with expected complex physics. The combination results in a globally fourth-order scheme in space and third order in time. Results confirming the method's high order of convergence are shown. Two-dimensional and three-dimensional test cases are presented and show good agreement with previous results. A simulation of Mach 3 flow over the logo of the Ubuntu Linux distribution is shown to demonstrate the method's capabilities for handling complex geometries. Results for Mach 6 wall-bounded flow over a three-dimensional cylindrical roughness element are also presented. The results demonstrate that the method is a promising tool for the study of hypersonic roughness-induced transition.

1. Introduction

The ability to accurately predict the location of laminar to turbulent transition in boundary layers is of great importance to the design of hypersonic vehicles [1]. In spite of its importance, the mechanisms leading to the transition of hypersonic boundary layers are still poorly understood. Predicting the location of transition is further complicated by the presence of surface roughness. Although a great deal of research has been done to study the effects of roughness on hypersonic transition, most of the data was for correlations to be used in the design of hypersonic vehicles. Very little is actually known about the exact physics behind hypersonic boundary-layer transition with roughness elements. The goal of this work was to develop a set of numerical tools for studying the effects of roughness elements on the transition of hypersonic boundary-layer flow. There are two major difficulties when simulating high-speed flow over roughness elements: imposing the roughness geometry and resolving the large variety of length scales present in the problem. The methods presented in this paper seek to address these difficulties.

* Corresponding author. Currently at the Lawrence Livermore National Laboratory.
E-mail address: greene@ucla.edu (P.T. Greene).

http://dx.doi.org/10.1016/j.jcp.2016.04.032
0021-9991/

1.1. Cartesian grid methods

The conventional approach to imposing geometries in direct numerical simulations (DNS) is to use a body-fitted grid [2]. In body-fitted grids, at least one of the edges of the computational domain will conform to the geometry. This makes the application of boundary conditions relatively simple. However, the generation of a body-fitted grid for complex geometries can be a very difficult and time-consuming process. As an alternative to body-fitted grids, an assortment of Cartesian grid methods have been developed [2]. The commonality between all the methods is that they can impose complex geometries on simple Cartesian grids. Most Cartesian grid methods fall into one of two categories: immersed boundary methods or sharp-interface methods.

The immersed boundary method was originally developed by Peskin [3,4]. Peskin used the immersed boundary method to study blood flow in a beating heart. The blood flow was simulated on a Cartesian grid while the elastic walls were discretized on a Lagrangian grid. The effect of the elastic wall was imposed on the fluid through a source term added to the equations governing the fluid flow. The source term would have an approximate Dirac δ function associated with it, which limits its area of influence to just near the walls. The use of a forcing term added to the governing equations is the defining characteristic of immersed boundary methods.

Goldstein et al. [5] extended the immersed boundary method for flows with immersed rigid walls. Goldstein et al. designed the forcing term to have a magnitude and direction opposing the local flow. This would drive the fluid to zero velocity at the immersed boundary. This method, and others like it, are usually referred to as continuous forcing methods. Mohd-Yusof [6] derived an alternative to this called the discrete forcing method. Mohd-Yusof changed the forcing term so that it would impose zero velocity at the boundary for all time.

The primary advantage of the immersed boundary method is the ease of its implementation. Since the body is imposed through a source term, the method can be easily incorporated into an existing code without having to alter the underlining solver. The numerical method used in the solver is simply applied everywhere, including inside any immersed bodies and across their surfaces. This has led to the immersed boundary method being used to simulate a large variety of problems such as modeling the cochlea [7], flow over an idealized road vehicle [8], and studying the dynamics of vesicles [9]. Although used successfully for many problems, the immersed boundary method has its disadvantages. Most applications of the immersed boundary method have been at low to moderate Reynolds numbers. At high Reynolds numbers, the spreading effect of the approximate δ function has a more detrimental effect on the flow since the local solution has greater importance [2]. In addition, the method is only locally first order, which may not have the required accuracy for some applications such as the numerical simulation of hypersonic boundary-layer transition.

While the immersed boundary method uses a forcing term added to the governing equation, sharp-interface methods usually alter the discretization of the governing equation near the body. The alteration allows boundary conditions to be applied just at the edge of the body. The methods generate precise locations for the body instead of smearing the interface as the immersed boundary method does. The two most common sharp-interface approaches are ghost fluid methods and cut-cell methods.

The ghost fluid method was developed by Fedkiw et al. [10,11]. The method begins by creating a layer of ghost fluid inside the immersed body. The values at the ghost points are extrapolated from the real fluid points in such a way that the boundary conditions are imposed at the interface exactly. Similar to the immersed boundary method, the interior discretization scheme is used across the body boundary, but now the values on the inside of the body are imposed by the method instead of advanced in time by the governing equations.

The primary motivation of the cut-cell method was to ensure global and local conservation of mass and momentum. To achieve this, the original formulation of the cut-cell method was made in a finite-volume discretization. Any cell that the edge of the body intersects is divided along the edge. The portion of the cells that are inside the body are discarded. If the center of the original cell is outside the body, then the remaining portion of the cell is used as the new cell. If the center is inside the body, then the remaining portion is merged into a neighboring cell. Unlike the previously mentioned methods, no values from inside the immersed body are ever used in the calculations. The cut-cell method was originally developed by Clarke et al. [12] for inviscid flow. Udaykumar et al. [13–15] and Ye et al. [16] extended the method to viscous flow.

Compared to the immersed boundary methods, sharp-interface methods tend to be more complex to implement. Sharp-interface methods are also victim to the "small cell problem" [17]. The small cell problem occurs when a grid cell becomes much smaller than its neighbors due to it being near the body's edge. For time dependent problems, this can severally limit the size of the permitted time step. Most sharp-interface methods have addressed this issue and provide a solution. The added complexity for sharp-interface methods is usually out weighted by the ability to precisely enforce the location of the immersed body's edge. This can be seen in the wide variety of problems the methods have been applied to such as shallow water flows [18], flapping flight [19], and magnetohydrodynamic flows [20].

1.2. Adaptive mesh refinement

Simulations of flow over roughness elements tend to involve a wide range of length scales. This is especially true for high-speed flows over roughness elements that have a height comparable to the boundary-layer thickness. When the flow encounters the roughness element, a new boundary layer is created along the roughness element. The thickness of this newly formed boundary layer can be more than an order of magnitude smaller than the thickness of the incoming boundary