Journal of Computational Physics 229 (2010) 7207–7237

Contents lists available at ScienceDirect

Journal of Computational Physics

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

A high-order cut-cell method for numerical simulation of hypersonic boundary-layer instability with surface roughness

Le Duan, Xiaowen Wang, Xiaolin Zhong *

Mechanical and Aerospace Engineering Department, University of California, Los Angeles, United States

ARTICLE INFO

Article history:
Received 1 July 2009
Received in revised form 3 June 2010
Accepted 7 June 2010
Available online 19 June 2010

Keywords:
High-order cut-cell methods
Cartesian grid methods
Surface roughness
Direct numerical simulation
Hypersonic boundary layer transition

ABSTRACT

Laminar-turbulent transition of hypersonic boundary layers can be affected significantly by the existence of surface roughness. Currently many important mechanisms of roughness-induced transition are not well understood. In recent years, direct numerical simulation (DNS) has been extensively applied for investigating instability and transition mechanisms of hypersonic boundary layers. Most of the past DNS studies, however, have been based on body-fitted grids for smooth surfaces without roughness. Due to complex geometry of embedded roughness, the use of body-fitted grids can be very difficult for flow with arbitrary surface roughness. In this paper, we present a new high-order cut-cell method to overcome the natural complexities in grid generation around arbitrary surface of roughness. The new method combines a non-uniform-grid finite-difference method for discrete grid points near the solid boundary and a shock-fitting method for the treatment of the bow shock. The non-uniform-grid finite-difference formulas are expressed in a general explicit form so that they can be applied to different multi-dimensional problems without any modification. The computational accuracy of new algorithms of up to $O(h^4)$ are tested on several one- and two-dimensional elliptic equations in irregular domains. In addition, the new method is applied to the simulation of the receptivity process of a Mach 5.92 flow over a flat plate under the combined effect of an isolated surface roughness element and surface blow and suction. A good agreement is found between the unsteady flow results and those obtained by a Linear Stability Theory (LST).

1. Introduction

The understanding of physical mechanisms of roughness induced boundary-layer transition is critical to the development of hypersonic vehicles [1]. Transition can have a first-order impact on the lift and drag, stability and control, and heat transfer properties of the vehicles [2]. For example, roughness induced transition is an important consideration in the design of thermal protection systems (TPS) of hypersonic vehicles [3,4]. Fig. 1 shows an example of surface roughness on test models for hypersonic boundary layer transition. For a reentry vehicle entering earth's atmosphere, it initially experiences a heating environment associated with a laminar boundary layer. As the vehicle altitude decreases, the vehicle surface becomes rougher and the boundary layer becomes turbulent. The transition from a laminar boundary layer to a turbulent one leads to the increase of surface heating rates by a factor of five or more. Thus the ability to understand and predict the physics of roughness induced transition plays an essential role in the design of TPS for reentry vehicles. Currently, surface roughness, especially arbitrary roughness induced laminar-turbulent transition in hypersonic boundary layers is still poorly understood due to the limitation in experimental facilities and numerical methods [5].

* Corresponding author.
E-mail address: xiaolin@seas.ucla.edu (X. Zhong).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.06.008

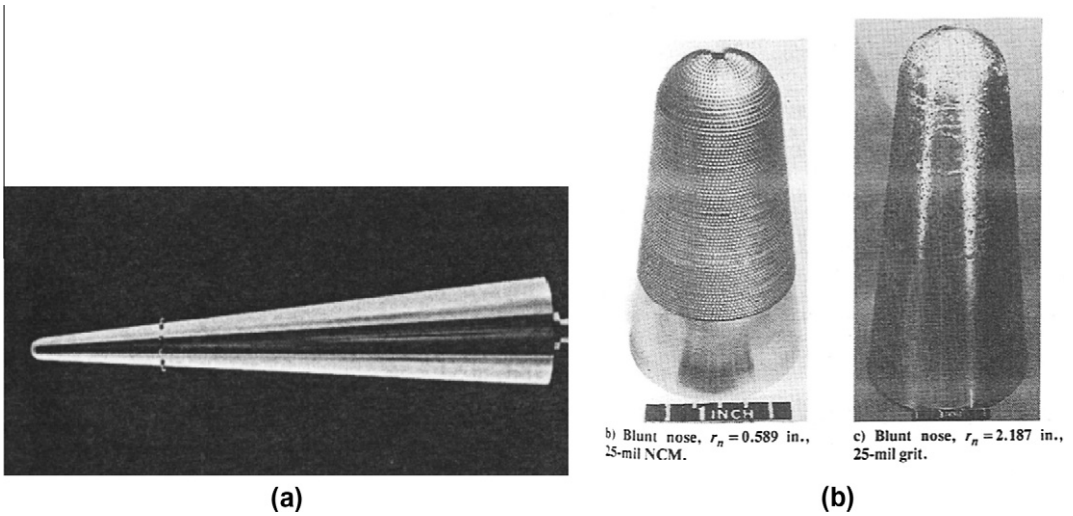

Fig. 1. An example surface roughness on test models for hypersonic boundary layer transition: (a) isolated roughness, (b) distributed roughness [6].

Direct numerical simulation has become an effective research tool for studying hypersonic boundary layer receptivity, stability and transition by numerically solving the time-dependent three-dimensional Navier–Stokes equations for the temporally or spatially evolving instability waves. It is necessary to use high-order numerical methods for the simulation in order to resolve the wide range of length and time scales of the complex wave fields in hypersonic boundary layers. Hence, high-order finite-difference schemes have recently received much attention for the direct numerical simulations of transitional and turbulent flows [7–12]. Zhong [13] presented a fifth-order upwind finite difference shock fitting method for the direct numerical simulation of hypersonic flows with a strong bow shock and with stiff source terms. The use of the shock-fitting method makes it possible to accurately compute the shock-disturbance interactions, and the development of instability waves in the boundary layers. The fifth-order shock-fitting schemes were derived on a uniform grid. For a curvilinear stretched grid, typically used in simulations of viscous flow in a boundary layer, the physical coordinates with a non-uniform grid are first transformed to a uniform grid in the computational coordinates. The high-order schemes are subsequently applied to the transformed equations in a uniform grid. The fifth-order shock-fitting scheme was used for numerical studies of the receptivity of two-dimensional Mach 15 flows over a blunt leading edge [14].

Most of the DNS methods, however, have been developed for smooth surfaces computed by body-fitted grids. Such methods may not be able to compute flow with surface roughness elements similar to those shown in Fig. 1. The main obstacle is the difficulty to generate smooth body-fitted grids around the surface of arbitrary roughness. One approach for overcoming the difficulty in grid generation is to use a Cartesian grid method, which is easy in grid generation. Cartesian grid methods can also take full advantage of fast computer architectures such as parallel computers and can serve as a very flexible method for simulating flow around complex geometries. On the other hand, in a Cartesian grid method, the grid lines are not aligned with the body surface. As a result, special treatment is needed to compute the flow equations in local regions adjacent to the solid surface.

Various Cartesian grid methods have been developed and used to solve problems with arbitrary geometry [15–19]. Peskin developed the immersed boundary method (IBM) [19] for the simulation of blood flow in hearts. The method is based on Cartesian grids where the surface of solid membrane is represented by a discrete delta function. This delta function is added into the Navier–Stokes equations to reflect the immersed boundary between the two phase flows. The resulting equations are discretized by a standard finite difference method in a fixed Cartesian grid system. Since its introduction, the IBM methods have been applied to many different fluid flow problems, including flow interaction with solid surface. For example, Marxen and Iaccarino [20] applied the IBM to simulate the effects of a localized two-dimensional roughness element on the disturbance amplification in a hypersonic boundary layer. However, since the immersed boundary method uses the discrete delta function approach, it leads to a smeared interface with a thickness in an order of a mesh width. The immersed boundary method is locally first-order accurate at the interface, which may not be accurate enough for the DNS of hypersonic boundary layer transition problems.

In contrast to the immersed boundary method with smeared interfaces, ‘‘sharp interface” Cartesian grid methods, which maintain second-order accuracy at the interfaces, have been developed [16,21]. Udaykumar et al. [16,17] used the finite-volume methodology to solve the incompressible Navier–Stokes equations for flow interacting with moving bodies. The method is based on a fixed Cartesian mesh where the solid boundaries can move across the grid lines. The flow equations are discretized by an overall second-order-accurate finite-volume technique. The interface is represented as a sharp boundary between the fluid and solid phases. A one-sided bilinear interpolation is applied to calculate the accurate flow conditions in the sharp irregular boundary. Johansen, McCorquodale and Colella [21,22] developed a similar sharp-interface Cartesian grid

method for solving two-dimensional Poisson and heat equations on irregular domains. Their method was based on finite-volume formulation, with imbedded irregular grid cells on the boundary. The irregular grid cells were treated by conservative differencing of second-order accurate fluxes on each cell volume. The method is second-order accurate for a problem with irregular boundary.

Fedkiw and co-workers [15,23] developed a high-order accurate finite-difference method to solve equations in a uniform Cartesian grid with an irregular domain. Their method is termed as ghost fluid method, and is based on the finite difference method on uniform grids. For the boundary points, they used the same stencil in finite difference discretization as the interior grids points, where the coefficients are computed by using high-order extrapolation in ghost cells outside the fluid domain. For 2-D or higher dimensional problems, they conducted the extrapolation in either a dimension by dimension manor, or in the normal direction directly, depending on the specific problem. Gibou and Fedkiw then applied their ghost fluid method to the heat equation and Stefan problem on irregular domains. To avoid the small cell problem, they showed that a good rule of thumb for removing small cell restriction is that the interpolation should be shift to be centered one grid points left or right if the interpolation stencil involves point considered to be relatively close to the boundary. The specific criterion on the smallness of the small cell is not stated in their paper. Gibou and Fedkiw [23] improved accuracy of the ghost fluid method to fourth-order accuracy for both the Laplace equation and the heat equation with Dirichlet boundary conditions on irregular domains. Finite difference discretization was used to solve the equations in an irregular domain. The sharp interface Cartesian grid method has also been termed cut-cell method in handling irregular grid cells along the boundary [24].

For multi-phase flows with moving interfaces, an immersed interface method was developed and has been widely used [25–35]. For an immersed interface method, a Cartesian grid is often used where a sharp interface moves freely across the fixed grid lines. This method can achieve a second- or higher-order global accuracy by incorporating jump conditions into the finite difference formulas for variables and their derivatives at the interface. This method was first developed by Leveque and Li [36] to solve elliptic equations with discontinuous coefficients and singular sources. The original method was second order in the interior of the domain and locally first order at the interface. Leveque and Li [37] subsequently extended the method to the Stokes flow with elastic boundaries or surface tension. Wiegmann and Bube [28,29] developed an explicit jump immersed interface method for special cases where the explicit jump conditions of variables and their high-order derivatives are known. This method can only achieve arbitrary high-order accuracy if the corresponding high-order derivatives of jump conditions can be analytically derived. Zhong [35] developed a new high-order immersed interface method which does not require second and higher derivatives of jump conditions. The main idea was to use a wider grid stencil across the interface instead of taking higher derivatives of jump conditions to achieve high-order accuracy for finite difference formulas at the interface.

For sharp-interface Cartesian grid methods, a "small cell problem" [38] of numerical instability would arise when finite-difference or finite-volume methods are applied to relatively small-sized irregular grid cells created by a sharp-interface Cartesian grid method. For a time-dependent problem, the small cell problem will significantly restrict the size of times steps in temporal integration methods. Many methods have been proposed to resolve the small cell problem. Berger and Leveque [38] used a rotating box method. Johansen and Colella [21] used a flux-redistribution procedure. Quirk [39] and Udaykumar et al. [17] employed a cell merging method to avoid small cells in order to maintain numerical stability.

Gibou and Fedkiw [23] presented a stable fourth-order finite difference method for solving the Laplace equation on an irregular domain. They showed that a good rule of thumb for removing small cell restriction is that the interpolation should be shift to be centered one grid points left or right if the interpolation stencil involves point considered to be relatively close to the boundary.

Due to the difficulty in grid generation for numerical simulation of high-speed flow with arbitrary surface roughness elements, it is advantageous to use a fixed-grid cut-cell method to compute such flow. The grid can be a smooth curvilinear body-fitted one along a baseline smooth surface without the roughness. The actual surface with roughness will cut across the grid lines, which create irregular grid points. Most of the previous Cartesian-grid cut-cell (or sharp-interface) methods with different boundary treatments, however, are only first- or second-order accurate at the interface. Low order schemes are adequate for most aerodynamic calculations, but they are not accurate enough for DNS of transition studies, where short wave-length shock/disturbance interactions need to be resolved with high accuracy [13]. There are only a few DNS studies of hypersonic boundary layers with roughness [20,40,41]. All of these studies used body-fitted grids. With the introduction of roughness, it is expected that the relative low accuracy at the interface may not be sufficient for numerically simulating laminar-turbulent boundary-layer transition involving surface roughness. In this paper, we propose a high-order cut-cell method up to 4th order based on finite difference method for the compressible Navier–Stokes equations. The generated grids are non-uniformly distributed near the boundary. Therefore, the numerical treatment in the interior grids points is similar to the ghost fluid method, but it is different in the boundary points. The coefficients are computed directly on the non-uniformly distributed stencil by using one-side discretization with explicit formula. For 2-D or higher dimensional problems, the cut-cell methods is conducted in a dimension by dimension manor. The new method is applied to solve the compressible Navier–Stokes equation in irregular domain. The shock interface is computed by a high-order shock fitting method. To overcome the small cell problem, we adopt a "drop point" approach. A critical ratio, defined as the minimum distance to the boundary, is used to classify the "drop point". The computational accuracy of our third and fourth order accurate cut-cell methods is tested for the computations of Poisson and hyperbolic equation with Dirichlet boundary conditions in irregular domains. We then test a third-order accurate cut-cell method for the two-dimensional Navier–Stokes equations. The new scheme is used to simulate steady and unsteady hypersonic boundary layer flows over a flat plate with an isolated surface roughness

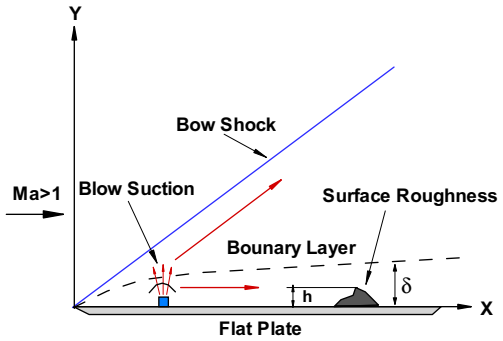

Fig. 2. An example hypersonic flow over flat plate with surface roughness induced boundary layer transition.

element (Fig. 2). The bow shock generated from the leading edge of the flat plate is treated as a computational boundary and discretized based on Zhong's [13] high-order finite-difference shock-fitting method. The unsteady flow over flat plate is excited by a narrow blowing and suction slot composed of 15 different frequencies mounted near the leading edge. Several multi-dimensional cases are tested to demonstrate the order of accuracy for these numerical methods.

The cut-cell method is inspired by the ghost fluid method of Gibou and Fedkiw [23] and several aspects of the methods are similar. On the other hand, the method is novel in several aspects of the method and in the complexity of the equations and problems which our methods are developed for and applied to. First, the new high-order cut-cell method is developed for much more complicated fluid flow problems of hypersonic viscous flow with a bow shock and surface roughness elements. The governing equations are full compressible Navier–Stokes equations with a shock wave, which is treated by a fitting method. On the other hand, Gibou and Fedkiw's method was developed for the linear Laplace and heat equations in a uniform Cartesian grid. Second, the numerical treatment near the boundary is different for these two methods. Non-uniformly distributed stencil is used in the cut-cell method, and the coefficients are computed directly by using one-side discretization with explicit formula. On the other hand, Gibou and Fedkiw use uniform grids in the computational domain, and obtain the coefficients by using extrapolation (i.e., the ghost fluid method). In addition, for 2-D and 3-D problems, Gibou and Fedkiw update the values in the ghost cell by using extrapolation in normal direction to the interface. Therefore, in their method the grids stencil in one direction may include the grids points in other directions. But our cut-cell method involves grids points in the finite difference direction only.

2. Governing equations

For direct numerical simulation of hypersonic boundary layer transition, the governing equations are the two-dimensional Navier–Stokes equations. We assume that we are dealing with Newtonian fluids with the perfect gas assumption and isothermal or adiabatic wall conditions. The governing equations can be written in the following conservation-law form in the Cartesian coordinates,

$$\frac{\partial U}{\partial t}+\frac{\partial F_j}{\partial x_j}+\frac{\partial F_{vj}}{\partial x_j}=0, \tag{1}$$

where U, F_j and F_{vj} are the vectors of flow variables, convective flux, and viscous flux in the jth spatial direction respectively, i.e.,

$$U=\{\rho,\rho u_1,\rho u_2,e\}, \tag{2}$$

$$F_j=\begin{Bmatrix}\rho u_j\\ \rho u_1u_j+p\delta_{1j}\\ \rho u_2u_j+p\delta_{1j}\\ (e+p)u_j\end{Bmatrix}, \tag{3}$$

$$F_{vj}=\begin{Bmatrix}0\\ \tau_{1j}\\ \tau_{2j}\\ \tau_{jk}u_k-q_j\end{Bmatrix}. \tag{4}$$

In this paper, only perfect-gas hypersonic flow is considered, i.e.,

$$p = \rho RT, \tag{5}$$

$$e = \rho\left(C_v T + \frac{1}{2}u_k u_k\right), \tag{6}$$

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) + \delta_{ij}\lambda\frac{\partial u_k}{\partial x_k}, \tag{7}$$

$$q_j = -k\frac{\partial T}{\partial x_j}, \tag{8}$$

where R is the gas constant. The specific heat C_v is assumed to be constant with a given ratio of specific heats γ. The viscosity coefficient μ can be calculated by Sutherland's law in the form:

$$\mu = \mu_r\left(\frac{T}{T_0}\right)^{3/2}\frac{T_0 + T_s}{T + T_s}, \tag{9}$$

where, for air, $\mu_r = 1.7894 \times 10^{-5}$ Ns/m^2, $T_0 = 288.0$ K, $T_s = 110.33$ K and λ is assumed to be $-2/3\mu$. The heat conductivity coefficient k can be computed through a constant Prantl number.

3. A high-order cut-cell method

Our high-order cut-cell method is presented in this section for the numerical simulation of hypersonic boundary-layer transition with arbitrary surface roughness. The traditional numerical methods of choice for the DNS of transitional and turbulent flows have been spectral methods because of their high accuracies [42–44]. But the applications of spectral methods have been limited to flows in simple domains. Finite-difference methods have recently received much attention for the DNS of transitional and turbulent flows, especially compressible flows [10,45–48], because they can be easily applied to complex geometries. High-order schemes are required because traditional second-order schemes do not provide adequate accuracy level for the direct numerical simulation. Most high-order finite-difference methods used in direct numerical simulation are central difference schemes [10,48] which introduce only phase errors but no dissipative errors in numerical solutions. The shortfall of central schemes is that they are often not robust enough in convection dominated hypersonic flow simulations. On the other hand, Rai et al. [45] show that upwind-bias schemes are very robust in hypersonic flow simulation even when they are made high-order accurate. They use a spatially fifth-order upwind finite-difference scheme in an upwind-bias stencil to compute the Navier–Stokes equations. The numerical dissipation in the upwind-bias schemes is enough to control the aliasing errors and could maintain the overall stability of the method.

The other issue that needs to be considered in DNS of hypersonic flow over blunt bodies is the treatment of shock waves. High-order linear schemes cannot be used for grid points across the bow shock fronts with steep gradients because spurious numerical oscillations are generated at the shocks. Many high-resolution shock capturing schemes, such as the WENO scheme [49], have been developed to capture shock waves as part of the numerical solutions without numerical oscillations. These shock capturing schemes are adequate for most aerodynamic calculations, but they are not accurate enough for DNS studies, where short wave-length shock/disturbance interactions need to be resolved with high accuracy. For DNS of hypersonic boundary layers, the bow shocks can be treated as a computational boundary using the shock-fitting method [50]. The use of the shock-fitting method makes it possible to apply high-order linear schemes for spatial discretization of the flow equations behind the bow shocks. Hussaini et al. [50] used the shock-fitting spectral method to simulate shock/turbulent interaction. To satisfy the accuracy requirement for capturing small disturbances inside the boundary layer for transition simulation, a high-order shock-fitting method is needed in order to compute shock interaction with high accuracy. Therefore, in this paper, our high-order cut-cell method is coupled with a fifth-order shock-fitting and upwind finite-difference scheme developed by Zhong [13] for computations of flow fields bounded by the bow shock and wall surface.

3.1. Computational grids and classification of grid points

A schematic of a computational domain and a cut-cell grid in roughness induced hypersonic boundary layer transition is shown in Fig. 3. This figure shows a typical hypersonic flow over a blunt body, where a bow shock is created by the supersonic freestream. In this paper, a high-order shock-fitting method is used to track the movement of the bow shock which is treated as the upper boundary of the computational domain. The computational grid for a shock fitting formulation is bounded between the bow shock above and the blunt body below.

The cut-cell grid is a smooth curvilinear grid fitted to the baseline body shape without the roughness. As a result, the roughness surface cuts across the grid lines. The roughness surface, Γ, is represented by surface equation in the following form,

$$\Gamma : f(x, y) = 0. \tag{10}$$

For a problem concerning practical arbitrary roughness in hypersonic vehicle surface, it is likely that there is not an analytical equation applicable to represent the shape of the roughness element. In this case, a set of n discrete coordinate points

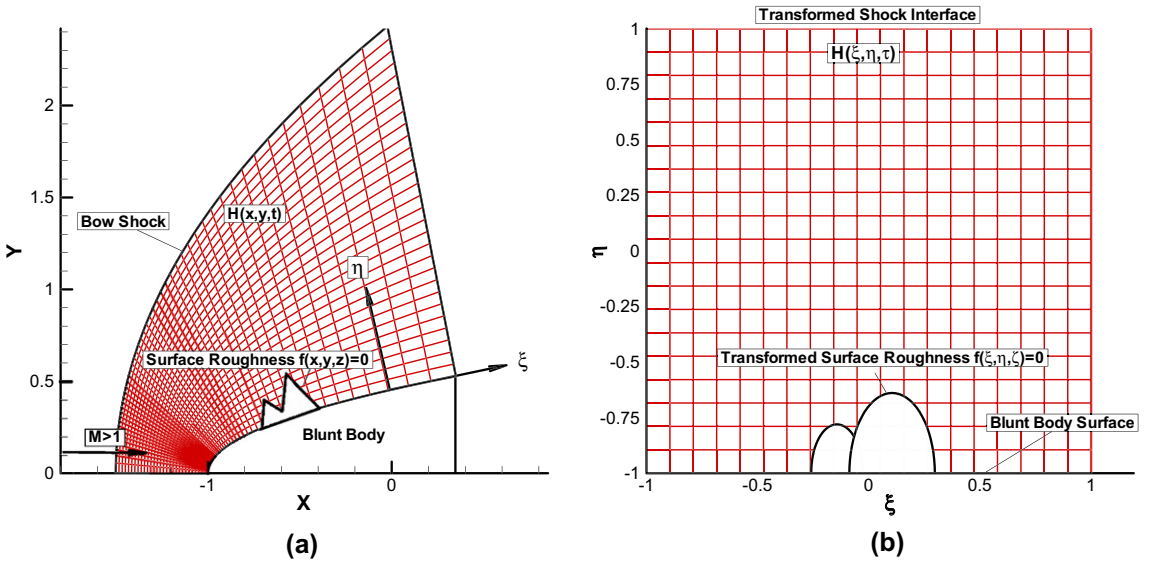

Fig. 3. Physical and computational domain and a cut-cell grid of hypersonic flow over a blunt body with surface roughness: (a) physical grid, (b) computational grid with a transformed roughness.

$\{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\}$ along the physical roughness surface are used to represent the surface. With these roughness data, we can reconstruct the roughness surface for simulation by using a high-order piecewise polynomial interpolation.

Both the governing equation (1) and the roughness equation (10) in the physical domain are transformed into a Cartesian computational domain bounded by bow shock and flat plate boundary. Under the computational coordinate system, the body fitted grids are represented by a curvilinear three-dimensional coordinates (ξ,η,τ) along the grid lines. The unsteady movement of the bow shock is treated as the computational upper boundary located at $\eta = \eta_{\max}$, which is time dependent. The other grid line $\xi = const$ remains stationary during computations. The coordinate transformation is defined by:

$$\begin{cases} \xi = \xi(x,y) \\ \eta = \eta(x,y,t) \\ \tau = t \end{cases} \leftrightarrow \begin{cases} x = x(\xi,\eta,\tau) \\ y = y(\xi,\eta,\tau) \\ t = \tau \end{cases} \tag{11}$$

where (x,y,t) are the physical coordinates defined under Cartesian coordinate system.

Substituting Eq. (11) into the governing equation (1), we obtain a system of transformed governing equations in the computational domain (ξ,η,τ) as

$$\frac{1}{J}\frac{\partial U}{\partial \tau} + \frac{\partial E'}{\partial \xi} + \frac{\partial F'}{\partial \eta} + \frac{\partial E'_v}{\partial \xi} + \frac{\partial F'_v}{\partial \eta} + U\frac{\partial (1/J)}{\partial \tau} = 0 \tag{12}$$

The transformed fluxes of the equation above are:

$$E' = \frac{F_1\xi_x + F_2\xi_y + F_3\xi_z}{J} \tag{13}$$

$$F' = \frac{F_1\eta_x + F_2\eta_y + F_3\eta_z + U\eta_t}{J} \tag{14}$$

$$E'_v = \frac{F_{v1}\xi_x + F_{v2}\xi_y + F_{v3}\xi_z}{J} \tag{15}$$

$$F'_v = \frac{F_{v1}\eta_x + F_{v2}\eta_y + F_{v3}\eta_z}{J} \tag{16}$$

where ξ_x, ξ_y, η_x, η_y are transformation metrics, and J is Jacobean matrix of coordinate transformation defined by

$$J = \frac{\partial(\xi,\eta)}{\partial(x,y)} \tag{17}$$

In addition to the transformation of the governing equations, the equation for the surface roughness is also transformed into the computational domain and can be represented as

$$f(\xi(x,y),\eta(x,y,t)) = 0 \tag{18}$$

The grids transformation metrics and the Jacobean matrix J can be obtained either by analytical formulas of the coordinate transformation or by numerical approximation.

With the coordinate transformation, a set of uniformly distributed Cartesian grids can be generated in the computational domain where the grid distribution in the physical domain is not uniformly distributed. Because smooth body-fitted grids are generated in the regular computational domain without the roughness as shown in Fig. 3(b), some of the Cartesian grid cells may be cut by the roughness boundary, which leads to irregular Cartesian grid cells.

In the current high-order cut-cell finite-difference method, four different types of grid points are defined according to their relative locations with respective to the roughness surface. Different numerical algorithms are implemented for different types of points. The four types of points, as shown in Fig. 4, are defined as:

- **Boundary points**: The points created by the intersection of roughness interface Γ and grid lines are boundary points. They are not part of the original Cartesian grid, and they are used in finite difference formulas for grid points adjacent to the roughness surface.
- **Irregular points**: For those points located close enough to the roughness boundary Γ that their finite difference stencils contain a boundary point, they are defined as irregular points. In the finite difference approximation involving an irregular point, local grid spacing becomes non-uniform because of the inclusion of a boundary point in the stencil.
- **Dropped points**: If a grid point is adjacent to a boundary point along a grid line with a distance smaller than a pre-specified critical ratio Θ in the ξ or η direction, it is defined as a dropped point along that direction. A dropped point is removed from the grid stencil in the formulation of a local finite difference approximation in the corresponding direction. It should be noted that a grid point can be a drop point in one direction, but a regular point in another direction. Therefore, dropped points are removed in the "dropped direction" only. The same points may be included in finite difference stencils in the other directions if these points are not defined as "dropped". The value of non-dimensional critical ratio Θ is an adjustable parameter. In the cases of third- and fourth-order methods, Θ is selected to be 0.5 and 1.0 respectively. All of the points on the solid side of the computational domain are defined as dropped points as well. They do not participate in any numerical calculation.
- **Regular points**: All other grid points produced by the intersection of grid lines themselves are defined as regular points. Since they are relatively far away from boundary points, a standard finite difference approach in a uniform grid can be applied.

3.2. Finite-difference algorithms for different types of grid points

The derivatives of the flux terms in Eq. (12) are discretized by different methods for the four different types of grid points. The flux terms in regular points are computed by the standard upwind finite difference scheme introduced in next section. To calculate the flux in irregular points, we use a high-order non-uniform-grid finite difference method, the stencil of which consists of regular, irregular and boundary points. The dropped points shown in Fig. 4 are not included in non-uniform or uniform grid stencils for computing flux terms in a "dropped" direction. The main purpose of defining a dropped point is to avoid the small cell problem introduced in Section 1. The removal of drop points from finite difference formulas ensures

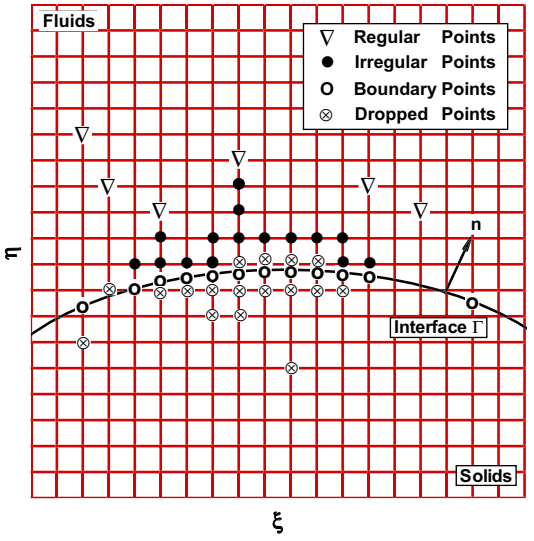

Fig. 4. Classification of four types of grids for the high-order cut-cell method in computational domain (ξ,η), where ∇ represents regular point, $\otimes$ dropped point, $\bigcirc$ boundary point, and $\bullet$ irregular point.

that the distance between two adjacent grid points in a finite-difference stencil is large enough in order to prevent from the deterioration of corresponding high-order method for computing the hyperbolic part of Eq. (12). Based on the different grid classifications in the computational domain, the details of the different discretization methods for the four types of points are described as follows.

3.2.1. Regular points and shock-fitting methods

Since regular points are not affected by the presence of roughness boundary, in this paper, regular high-order upwind schemes of Zhong [13] are used for the discretization of the governing equations for these points. In addition, the bow shock shown in Fig. 3 is treated by a high-order shock-fitting approach. The finite difference methods and formulas used for the regular points and the bow shock are briefly described here.

In the discretization of the Navier–Stokes equations (Eq. (12)) at a regular point, spatial derivatives in the streamwise (ξ) and wall-normal (η) directions are discretized by fifth-order finite difference schemes. In a shock fitting scheme, the shock front is represented by a shock height function, $H = H(\xi, \tau)$, which is defined as the distance along a wall normal grid line between the wall surface and the shock (Fig. 3). Function H is not known in advance, and it is governed by two additional equations of shock velocity and acceleration. The flow variables for grid points immediately behind the shock are determined by the Rankine–Hugoniot relations across the shock and a characteristic compatibility equation behind the shock. From these two equations, the velocity and acceleration equations of the shock front can be obtained as:

$$\begin{cases} \frac{\partial H}{\partial \tau} = H_\tau \\ \frac{\partial H_\tau}{\partial \tau} = H_{\tau\tau}\left(\xi, U_s, \frac{\partial U_s}{\partial \tau}, H, H_\tau, U_\infty, \frac{\partial U_\infty}{\partial \tau}\right) \end{cases} \tag{19}$$

where $H_{\tau\tau}\left(\xi, U_s, \frac{\partial U_s}{\partial \tau}, H, H_\tau, U_\infty, \frac{\partial U_\infty}{\partial \tau}\right)$ is a relation determined by flow conditions and their time derivatives on both sides of the shock, subscripts s and ∞ represent flow variable at the shock on the high-pressure and freestream sides of the shock. In a time marching computation, an initial shock and flow field behind the shock is given. As the computations progress in time, the shock shape as well as the flow variables in the physical domain are updated in each time step according to a time step scheme to solving Eqs. (12) and (19). More details of the derivation of the shock fitting formulas and numerical methods can be found in Zhong [13].

The transformed equation (12) is discretized by high-order finite difference methods, two sets of difference schemes are employed to calculate viscous and inviscid flux terms separately. For the inviscid flux terms, upwind schemes of up to fifth-order combined with a local Lax–Friedrichs (LLF) scheme are used, while central schemes of up to sixth-order are used to discretize the viscous flux derivatives. Specifically, an inviscid flux term F' in Eq. (12) can be divided into two parts with pure negative and positive eigenvalues as

$$F' = F'^{+} + F'^{-} \tag{20}$$

where

$$\begin{cases} F'^{-} = \frac{1}{2}(F' - \lambda U) \\ F'^{+} = \frac{1}{2}(F' + \lambda U) \end{cases} \tag{21}$$

The parameter λ is chosen as a sufficient large so that F'^{-} and F'^{+}contain only pure negative and positive eigenvalues, respectively. The specific form of λ used in this paper is:

$$\lambda = \frac{|\nabla \eta|}{J}\left(\sqrt{(\varepsilon c)^2 + u'^2} + c\right) \tag{22}$$

where c is the local speed of sound, ε is a small dimensionless parameter which introduce smoothness of the flux splitting, and u' can be calculated by

$$u' = \frac{\eta_x u + \eta_y v + \eta_t}{|\nabla \eta|} \tag{23}$$

Since F'^{-} and F'^{+}contains only negative and positive eigenvalues respectively, a fifth-order explicit upwind scheme is used to discretize their derivatives in order to improve the overall computational stability. A fifth-order finite difference formula for the first-order derivative of F'^{+}can be written as

$$\frac{\partial F'^{+}}{\partial \xi} = \frac{1}{\Delta h}\sum_{k=-3}^{3} a_{i+k} F'^{+}_{i+k} - \frac{\alpha}{6!}\left(\frac{\partial F'^{+}}{\partial \xi}\right)_i^6 + \cdots \tag{24}$$

where Δh is the size of spatial grids. The coefficients of a seven-point stencil can be calculated by using a Taylor expansion as follows:

$$\begin{cases} a_i = \frac{1}{60}\left(-\frac{5}{3}\alpha\right) \\ a_{i\pm1} = \frac{1}{60}\left(\pm 45 + \frac{5}{4}\alpha\right) \\ a_{i\pm2} = \frac{1}{60}\left(\mp 9 - \frac{1}{2}\alpha\right) \\ a_{i\pm3} = \frac{1}{60}\left(\pm 1 + \frac{1}{12}\alpha\right) \end{cases} \tag{25}$$

The parameter α is a free parameter. When $\alpha < 0$, Eq. (24) represents a fifth-order upwind schemes used for the inviscid flux term F'^{+}. The specific value of α is chosen based on a stability analysis for the high-order finite difference method in [13]. On the other hand, when $\alpha = 0$, the scheme is a sixth-order central difference scheme, which is used to discretize the viscous terms.

Combined with finite difference treatments for the other types of grid points, the spatial discretizations of Eqs. (12) and (19) lead to a system of ordinary differential equations for flow variables, shock velocity and shock acceleration. For steady and unsteady flow problems, a Runge–Kutta scheme is used to discretize the system of equations.

3.2.2. Irregular points

Contrary to the standard fifth-order finite-difference schemes used for regular points, special treatment is needed in the discretization of the governing equations for the irregular points because boundary points are included into the finite difference stencil. Fig. 5 shows a schematic of a grid stencil for irregular points near the boundary. In this figure, grid points ξ_2 and ξ_3 are irregular points, ξ_1 is a boundary point, while the grid point between ξ_1 and ξ_2 is a dropped point because it is too close the boundary point. The dropped point is removed from the stencil for the irregular point schemes. The rest of the grid points are regular points. For example, for a local third-order finite-difference approximation at the irregular point ξ_2, the grid stencil consists of the following five grid points: $\xi_1, \xi_2, \ldots, \xi_5$. The grid spacing between the points involved are not uniform because

$$\sigma = \frac{\theta}{\Delta h} \neq 1 \tag{26}$$

where θ is the spacing between ξ_1 and ξ_2, and Δh is the uniform grid spacing of the regular grid. Therefore, a non-uniform-grid finite difference schemes are needed for irregular points.

It is assumed that p is the local order of accuracy in boundary of the simulation. In order to maintain a $(p+1)$th order global accuracy for the discretization of both the inviscid and viscous flux terms, all local non-uniform schemes for irregular points need to be at least pth order accuracy. In our construction of high-order cut-cell scheme, if a grid stencil with a number of q grids is used for discretizing regular points near the boundary in each direction of the computational domain, there are $\lfloor q/2 \rfloor$ irregular points near a boundary surface. Fig. 5 shows the case of a third-order scheme with $q = 5$, where there are two irregular points ξ_2 and ξ_3. We denote the collection of all irregular points near this boundary in one direction as set Ω_p^{ξ}, where the superscript represents the direction of the stencil and the subscript represents the local order of schemes at this point.

As a general convention, the boundary point is labeled as ξ_1, the irregular points are labeled as $\xi_2, \xi_3, \ldots, \xi_{\lfloor q/2 \rfloor + 1}$ sequentially in the order of their distances from the boundary point. A special case with $p = 3$ and $q = 5$ is shown in Fig. 5. There may or may not be a drop point in the stencil for an irregular point depending on the spacing between the boundary point and its closet regular grid point. As discussed earlier in this paper, if the non-dimensionalized grid spacing between the dropped point $\otimes$ and the boundary point $\bigcirc$ is less than a pre-described critical ratio Θ, this dropped point is removed from the grid stencil in corresponding direction. Otherwise, there is no dropped point in the stencil.

3.2.2.1. Finite difference formulas for viscous flux terms in irregular points. The discretization of viscous term F_v for an irregular point involves the calculations of second order derivatives since F_v contains gradient terms, such as $\nabla\vec{u}$ and ∇T. Under the coordinate transformation of Eq. (11), $\nabla\vec{u}$ and ∇T can be expressed as derivatives with respect to coordinates in the computational domain $(\xi, \eta, \varsigma, \tau)$. A two-step algorithm is employed for the computations of viscous fluxes. First, the gradient of velocity and temperature $\nabla\vec{u}$ and ∇T is computed at each grid point. Using these results, the values of flux vector F_v is evaluated at all grid points. Second, the derivatives of F_v are computed by a finite difference scheme. Since the computations in both steps involve only calculations of first-order derivatives, the same high-order finite-difference schemes are used at each step. In order to do this, the first derivative has to be computed at the boundary points. For boundary points with regular

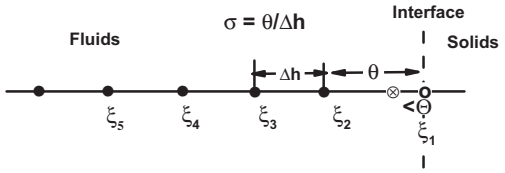

Fig. 5. A schematic of a grid stencil for an irregular point with $p = 3$, $q = 5$ and $\Omega_3^{\xi} = \{\xi_2, \xi_3\}$, θ is non-uniform grid spacing after removing a dropped point, Δh is the normal grid spacing, $\otimes$ represents the dropped point, $\bigcirc$ represents the boundary point, and $\bullet$ represents irregular and regular points.

finite-difference stencil, the first-derivative can be calculated by using one-side finite-difference scheme, e.g. for 4th order simulation:

$$(u_x)_1 = \frac{4u_2 - 6u_3 + 4u_4 + u_5}{\Delta h} \tag{27}$$

For boundary points with irregular finite-difference stencil, the first-derivative can be evaluated by using non-uniform one-side finite-difference scheme. The derivation details can follow the following steps.

The general formulation of a non-uniform-grid finite-difference scheme for computing the viscous terms for an irregular point ξ_i of Ω_p^ξ can be written in the following form:

$$\left(\frac{\partial F'_v}{\partial \xi}\right)_i = \frac{1}{\Delta h}\sum_{k=1}^{q} a_{i,k}(\sigma) F'_{v,k} - \frac{C(\Delta h)^{q-1}}{q!}\left(\frac{\partial F'_v}{\partial \xi}\right)_i^{q-1} \quad \text{for } \xi_i \in \Omega_p^\xi \text{ where } i = 2,3,\ldots,\lfloor q/2 \rfloor + 1 \tag{28}$$

The subscripts of coefficient $a_{i,k}$ stand for the kth coefficient for the ith irregular point as defined for the case of $p = 3$ and $q = 5$ (Fig. 5). The coefficient $a_{i,k}$ is function of σ, which is defined in Eq. (26), and C is a constant.

The coefficients of non-uniform-grid finite-difference equation (Eq. (28)) for every ith irregular grid point in set Ω_p^ξ can be derived either by a Taylor series expansion, or by taking a derivative of a polynomial interpolated through the non-uniform stencil. We use polynomial interpolation in this paper. Specifically, for each irregular point, the Lagrange interpolation polynomial can be written as

$$\widetilde{P}_{q-1}(\xi) = \sum_{l=1}^{q}\left(\prod_{m=1,m\neq l}^{q}\frac{\xi - \xi_l}{\xi_l - \xi_m}\right) F'_{v,l} \tag{29}$$

where $\widetilde{P}_{q-1}(\xi)$ is a polynomial interpolating through the grid stencil. Differentiating Eq. (29) once with respect to ξ, we have

$$\frac{\partial F'_v}{\partial \xi} \approx \frac{\partial \widetilde{P}_{q-1}(\xi)}{\partial \xi} = \sum_{l=1}^{q}\left(\sum_{n=1,n\neq l}^{q}\left(\prod_{m=1,m\neq l,m\neq n}^{q}\frac{\xi - \xi_m}{\xi_l - \xi_m}\right)\frac{1}{\xi_l - \xi_n}\right) F'_{v,l} \tag{30}$$

Substituting $\xi = \xi_i$ into the equation above and comparing terms with those of Eq. (28), we obtain the coefficients $a_{i,k}$ as follows

$$a_{i,k} = \sum_{n=1,n\neq k}^{q}\left(\prod_{m=1,m\neq k,m\neq n}^{q}\frac{\xi_i - \xi_m}{\xi_k - \xi_m}\right)\frac{1}{\xi_k - \xi_n} \quad \text{for } k = 1,2,\ldots,q \tag{31}$$

where $a_{i,k}$ is a function of σ defined in Eq. (26). In a shock-fitting calculation, the computational grid changes with the movement of the bow shock. As a result, the value of θ is a function of time because of the shock movement. The explicit formulas of $a_{i,k}$ for the non-uniform-grid algorithm for the discretization of the viscous terms are listed in Tables A1–A3 in Appendix.

During the time advancement in the simulation, the grid metrics in the computational domain bounded by the moving shock vary from one time step to the next. As a result, the σ values used in Tables A1–A6 are not constant. Thus we need to recalculate the finite-difference coefficients $a_{i,k}$ for all irregular points in each time step. In our simulation, the formulas for the coefficients in Tables A1–A6 are stored in the computer memory, and their values are computed explicitly when the finite-difference schemes are implemented in each direction.

3.2.2.2. Finite difference formulas for inviscid flux terms in irregular points. Similar to the local algorithms for the viscous terms at irregular points, a non-uniform-grid high-order upwind scheme is used to discretize the inviscid fluxes, $F'^{+}$and F'^{-}, as defined in Eq. (20). For every irregular grid point, there are several possible grid stencils for finite-difference approximation of the flux derivatives of the same accuracy order. Different choices of stencils for these boundary closure schemes lead to different stability characteristics for the overall algorithm. Since F'^{+} and F'^{-} have either all positive or all negative eigenvalues, local grid stencils for finite-difference approximation of the flux derivatives are chosen so that the discretization for the fluxes are upwind biased, while maintaining high-order accuracy. Therefore, for a given irregular grid point, the stencil for F^{+} may be different from the stencil for F'^{-}. In addition, a grid stencil for these fluxes may or may not include the boundary point.

The non-uniform-grid finite-difference schemes for the inviscid flux terms of positive and negative eigenvalues can be written as

$$\left(\frac{\partial F'^{+}}{\partial \xi}\right)_i = \frac{1}{\Delta h}\sum_{k=1}^{q} b_{i,k}^{+} F'^{+}_k \quad \text{for } \xi_i \in \Omega_p^\xi \text{ where } i = 2,3,\ldots,\lfloor q/2 \rfloor + 1 \tag{32}$$

$$\left(\frac{\partial F'^{-}}{\partial \xi}\right)_i = \frac{1}{\Delta h}\sum_{k=1}^{q} b_{i,k}^{-}(\sigma) F'^{-}_k \quad \text{for } \xi_i \in \Omega_p^\xi \text{ where } i = 2,3,\ldots,\lfloor q/2 \rfloor + 1 \tag{33}$$

The stencil contains a total of q grid points as shown in Fig. 5. The subscripts of coefficients $b_{i,k}^{+}$ and $b_{i,k}^{-}$ stand for the kth coefficient for the ith irregular point as defined in Fig. 5. The upwind schemes are represented by different sets of coefficients of

the two formulas above. Because the upwind bias stencil used for $\frac{\partial F'^{+}}{\partial \xi}$ does not include the boundary point, the coefficients $b^{+}_{i,k}$ is not a function of σ defined in Eq. (26). On the other hand, $b^{-}_{i,k}$ is a function of σ because the upwind stencil for $\frac{\partial F'^{-}}{\partial \xi}$ includes the boundary point. The coefficients for each upwind scheme above can be calculated following the same general formulas given by Eqs. (30) and (31), and they are listed in Tables A4–A6, in Appendix.

The higher-order non-uniform finite-difference stencils require sufficiently many grid points near the roughness surface. When the grids are insufficient for local high-order finite-difference stencil, there are two approaches to overcome the problem. The first approach is to maintain the original number of grids point in the local high-order finite-difference stencil. More grid points can be clustered or generated near the irregular boundary by either adjusting the coordination transformation, or refining the grids in the computational domain. The global grids refinement tends to be very computationally expensive to later simulation. Local grids refinement approach can also be adapted, but extra interpolation procedure has to be implemented for different levels of grids. By generating enough grids points near the irregular interface, the order of accuracy for cut-cell method is maintained. The second approach is to reduce the number of grids in the local finite-difference stencil, where corresponding lower order of non-uniform finite-difference method can be applied. By using this treatment, the local order of accuracy is reduced. If the boundary nearly degenerates, the accuracy of current cut-cell method will not be affected, provided that the boundary interface can be expressed accurately in a numerical or analytical form.

3.2.3. Boundary points

As shown in Fig. 4, boundary points are the marker points located at the intersection between the roughness surface and grid lines. The governing equations are not solved at these points. However, flow variables at these points are needed for finite-difference algorithms of the irregular points near the roughness surface. The solid interface imposes the non-slip and non-flow-through conditions for flow velocities at the boundary points. Depending on the actual thermal conditions of the solid surface, either an isothermal or adiabatic wall can be specified. In this paper, only the isothermal wall is considered. In this case, both the velocities and temperature of the boundary points are specified by the wall boundary conditions. The local pressure on the solid surface (boundary points) needs to be computed by the local flow conditions near the wall.

There are several approaches to compute the pressure at the boundary points. One approach is to integrate a local wall-normal momentum equation to obtain the wall pressure. We can also use an approximation assumption of zero pressure gradients at the wall to determine the pressure there. In a previous paper, Zhong used a fifth-order polynomial extrapolation to determine the wall pressure [13]. Satisfactory results have been obtained with this approach. We follow the extrapolation approach of Zhong [13] to determine the pressure at the boundary points.

In computing pressure at the boundary points, the polynomial extrapolation is required to have comparable order of accuracy as that of the interior schemes in order to maintain the expected global accuracy. To maintain a $(p+1)$th order global accuracy of the upwind schemes developed in the last sections, it is desirable to have at least pth order of accuracy for the extrapolation for the boundary points.

Since a boundary point is formed by the intersection of the roughness interface with one of the grid lines, the extrapolation is conducted along the direction of the same grid line. We use the grid line along the ξ direction as an example. The methods can be applied to other directions similarly. In two- and three-dimensional cases, there is options of either constructing the extrapolation along the direction normal to the solid interface, or doing it along the grid lines. Though either method can be used in multi-dimensional problems, the latter approach is used in this paper since it is simper in implementation and more stable for simulations of current tests. In this case, the determination of pressure at the boundary points is a one-dimensional extrapolation along one of the grid lines. The one-dimensional stencil of Fig. 5, which involves non-uniform grid spacing θ, is used to derive the extrapolation formulas for the pressure at the boundary point. For example, for the case of four point extrapolation shown in Fig. 5, pressure at the boundary point $p(\xi_1)$ is obtained by a third degree polynomial interpolating through the following four interior pressures: $p(\xi_2)$, $p(\xi_3)$, $p(\xi_4)$, and $p(\xi_5)$. The grid spacing between neighboring grid points is a constant value of Δh, with the exception that the distance between the first and second points are θ. If there is a dropped point in the stencil, the dropped point is not used in the extrapolation calculations.

For two-dimensional problems, the high-order extrapolation along the ξorη direction is employed to extrapolate pressure from the interior domain into the boundary. A total of p grid points in the direction associated with the boundary point are chosen to be included in the extrapolation stencil. Again, dropped points are not included in the extrapolation stencil. This procedure can prevent the small cell problem of producing numerical instability near the solid boundary.

3.2.4. Dropped points

In this paper, in order to avoid the small cell problem, a grid point is designated as a dropped point if its distance from a neighboring boundary point along a grid line is smaller than a pre-determined value Θ. The dropped point, which is associated with a grid direction, is removed in the finite-difference grid stencil along the grid line for the irregular points near the boundary. On the other hand, a grid point may become a dropped point in one direction, but remain a regular or irregular point in another. For example, point P_1 in Fig. 6 is a dropped point in the ξ direction, but a regular point in the η direction. In this case, the flow variables at point P_1 are not used in finite difference formulas for derivatives in the ξ direction. However, the flow variables at the same grid point are needed for finite difference formulas for derivatives in the η direction. The flow variables at this dropped point P_1 are obtained by the interpolation of a stencil along the ξ direction.

For a grid point which is a regular or irregular point in one direction, but a dropped point in another, finite difference schemes along the former direction may include this point in its stencil. As shown in Fig. 6 for the case of $p = 3$, the

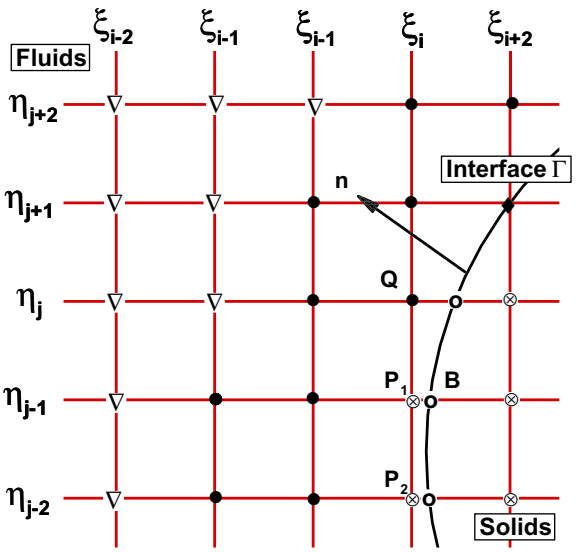

Fig. 6. An example of a grid point, such as point P_1 and point P_2, which is a regular or irregular point in one direction, but a dropped point in another for the case of $p = 3$: $\triangledown$ represents regular points, $\otimes$ dropped points, $\bigcirc$ boundary points, and $\bullet$ irregular points.

finite-difference stencil for an irregular point Q located at (ξ_i,η_j) contains five points in the η direction, which are in set $\Omega_Q = \{(\xi_i,\eta_{j+2}),(\xi_i,\eta_{j+1}),(\xi_i,\eta_j),(\xi_i,\eta_{j-1}),(\xi_i,\eta_{j-2})\}$. There are two dropped points along the ξ direction in this set of stencil Ω_Q: point P_1 at (ξ_i,η_{j-1}) and point P_2 at (ξ_i,η_{j-2}). If the points P_1 and P_2 are removed from the stencil used in the calculations of flux terms $\partial F'/\partial\eta$ and $\partial F'_v/\partial\eta$ in Q, the stencil set Ω_Q needs to be shifted two grids down to include (ξ_i,η_{j-3}) and (ξ_i,η_{j-4}) to maintain the accuracy. In this case, the resulting stencil for Q may contain a significantly large interval θ compared with the normal grid spacing Δh, which may lead to a deterioration of accuracy of the method. Therefore, we maintain the original grid stencil Ω_Q, which includes points P_1 and P_2, along the η direction for point Q. We calculate the flow variables of these two dropped points by interpolation along the ξ direction.

For the case of $p = 3$ for point P_1, a third-order polynomial interpolation along the ξ direction is employed to compute the flow variables at this point. The interpolation is carried out along the ξ direction, which is along the direction where the point is dropped. For $p = 3$ as shown in Fig. 6, the interpolation stencil for point P_1 is set $\Omega_P = \{(\xi_{i-1},\eta_{j-1}),(\xi_{i-2},\eta_{j-1}),B\}$, where B represents the boundary point. For higher order method, the order of interpolation needs to be increased accordingly. For $p = 4$, a fourth order interpolation should be used. For a general case of pth order methods at the boundary, a total of $p-1$ adjacent grid points and exactly one boundary point along the ξ direction are chosen as the interpolation stencil. The interpolant can be written as

$$\widetilde{U}(\xi_{p_1}) = \sum_{n=1}^{p}\left(\prod_{l=1,l\neq n}^{p}\frac{\xi_{p_1}-\xi_l}{\xi_n-\xi_l}\right)U_n \tag{34}$$

where ξ_{p_1} is the ξ coordinate of dropped point, $\{U_i,\ i = 1,\ldots,p\}$ is conservative flow variables at the grid points of the interpolation stencil.

Similar interpolation procedures can also be carried out if a point is designated as a dropped point in the η direction, but is an irregular or regular point in the ξ direction. If a grid point is designated as a dropped point in both the ξ and η directions, there is no need to do interpolations because this point is removed from the calculations of both directions.

4. Numerical results

4.1. One-dimensional boundary value problem in an irregular domain

As a first test case of our new high-order cut-cell method, the following one-dimensional boundary value problem in an irregular domain is considered:

$$T_{xx} = f, \quad x \in [0, C_\delta] \tag{35}$$

where

$$f = \beta\sin(\alpha x)$$

Eq. (35) is defined in an irregular domain $\Omega = [0, C_\delta]$ and Dirichlet boundary conditions are imposed as: $T(0) = 0$, $T(C_\delta) = 1$, where C_δ is a constant smaller than 1.0. A set of uniform N grid points are used to cover the regular computational domain $\Omega' = [0,1]$. As a result, the right boundary of domain Ω is not located at one of the regular grid points. The computational accuracy of our cut-cell method is tested in this problem with the irregular grid points. The specific parameters used in our computational tests are:

$$\begin{cases} \alpha = 7\sqrt{3}\pi \\ \beta = -147\pi^2 \\ C_\delta = \sqrt{3}/2 \end{cases}$$

The exact solution can be written as,

$$T_e = \sin\left(7\sqrt{3}\pi x\right) \tag{36}$$

A set of uniform N grid points shown in Fig. 7 are generated to cover the regular computational domain of $\Omega' = [0,1]$ to test the order of accuracy of the cut-cell method. The grid size is $\Delta h = 1/N$ with the grid nodes x_i for located at their center. For a given value of C_δ, the boundary point is located at $x = C_\delta$, the number of grid points which are located on the left side of the boundary point is n. There is a dropped point immediately on the left of the boundary point if the distance between them is less than a critical value of $\Theta\Delta h$. In this case, we need to compute T_i for $i = 1, \ldots, (n-1)$. If we set the order of accuracy of the method to be fourth-order for regular points and third-order for irregular points near the boundary, the classification of the grid points is:

$$\begin{cases} i = 1, \ldots, (n-3) & \text{regular points} \\ i = n-2, n-1 & \text{irregular points} \end{cases} \tag{37}$$

The grid spacing between the boundary point and the rightmost irregular point at $i = n-1$ is

$$\theta = C_\delta - (n-1)\Delta h \tag{38}$$

And the parameter σ used in the non-uniform-grid algorithms can be calculated by

$$\sigma = \frac{\theta}{\Delta h} \tag{39}$$

The solution to Eq. (35) is computed at the grid nodes and is written as

$$T_i = T(x_i) \quad (i = 1, \ldots, n-1) \tag{40}$$

For regular points, a fourth-order discretization is considered, which can be written in the form,

$$\begin{aligned} (T_x)_i &= \frac{5T_{i-2} - 40T_{i-1} + 40T_{i+1} - 5T_{i+2}}{60\Delta h} \quad (i = 2, \ldots, n-3) \\ (T_x)_{i=1} &= \frac{-3T_0 - 10T_1 + 16T_2 - 6T_3 + T_4}{12\Delta h} \end{aligned} \tag{41}$$

To consistent with our method for the Navier–Stokes equations, the second derivative $(T_{xx})_i$ is obtained by applying the first-derivative formulas above twice.

For the irregular points, we let grid stencil be $q = 5$ and use a third-order construction described in Section 3.2 for the two irregular points. To prevent potential deterioration of accuracy for the high-order finite-difference method in the surrounding points, the first irregular point may be defined as a dropped point depending on its distance from the boundary point. After the irregular points and grid stencil have been identified, fourth-order non-uniform-grid finite-difference schemes are used for the irregular points, i.e.,

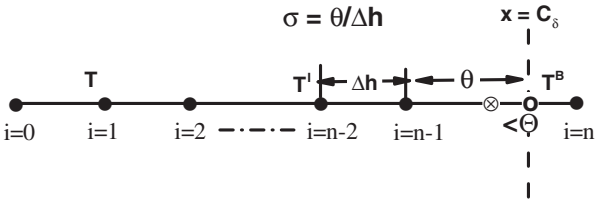

Fig. 7. Computational grid used for the 1-D test case with $p = 3$, $q = 5$ and $\Omega_3^\xi = \{\xi_{n-2}, \xi_{n-1}\}$, θ is the non-uniform grid spacing after removing a dropped point, where Δh is the normal grid spacing, $\otimes$ represents the dropped point, $\bigcirc$ represents the boundary point, and $\bullet$ represents irregular and regular points.

$$\left(T_x^I\right)_{n-2} = \frac{a_{3,5}T_{n-4} + a_{3,4}T_{n-3} + a_{3,3}T^I_{n-2} + a_{3,2}T^I_{n-1} + a_{3,1}T^B}{\Delta h} \tag{42}$$

$$\left(T_x^I\right)_{n-1} = \frac{a_{2,4}T_{n-3} + a_{2,3}T^I_{n-2} + a_{2,2}T^I_{n-1} + a_{2,1}T^B}{\Delta h} \tag{43}$$

where T^I is the irregular point, T^B is the boundary point. The coefficients $a_{i,j}$, which are function of σ, can be found in Table A2.

The finite-difference non-uniform-grid schemes given by Eqs. (41)–(43) results in a system of linear equations for T_i with a sparse coefficient matrix. A fast iteration method or a direct Gaussian Elimination method can be employed to obtain the numerical solution of this one dimensional equation.

Fig. 8 compares the exact solution and the solution of the fourth-order cut-cell method for the case of $N = 320$. A good agreement is achieved between the two sets of solutions. Table 1 gives the computational errors, in L^∞ and L^2 norms, of the numerical solutions with different sets of grids. To evaluate the computational order of accuracy of the method using the grid refinement approach, numerical results for two sets of grids with N and $N/2$ are compared. When the number of grids is increased by a factor of 2, the errors are expected to decrease by a factor of 16 for a fourth-order method. An error ratio is defined as

$$Ratio = \frac{\|E_{N/2}\|_\infty}{\|E_N\|_\infty} \tag{44}$$

where the infinite norm is used for error calculations. The order of accuracy can be then calculated as

$$p = \frac{\ln(\|E_{N/2}\|_\infty / \|E_N\|_\infty)}{\ln(2)} \tag{45}$$

Because we use a locally third-order ($q = 5$) stencil for the irregular point at $i = n - 1$ as stated in Eq. (43), the order of overall accuracy is below 4 under L^∞ norm. But Table 1 shows that the global order of accuracy based on the L^2 norm does approach 4 for the current cut-cell method.

In order to evaluate the effects of boundary closure schemes for the irregular points, we increase the order of the schemes for the irregular points to fourth order, while the fourth-order schemes for the regular points remain the same. The results

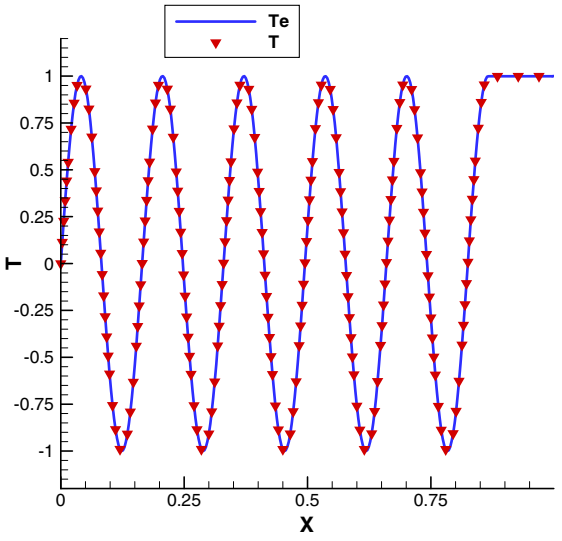

Fig. 8. Comparison of numerical solution $T(x)$ computed by the fourth-order cut-cell method and the exact solution $T_e(x)$ ($N = 320$).

Table 1
Numerical errors of the fourth-order cut-cell method for computing the one-dimensional boundary value problem with an irregular domain.

N	Θ	$\|E_N\|_\infty$	Ratio	p	$\|E_N\|_2$	Ratio	p
20	0.25	1.521(0)			8.808(−1)		
40	0.25	6.001(−2)	25.4	4.66	1.720(−2)	51.2	5.67
80	0.25	5.166(−3)	11.6	3.53	1.136(−3)	15.1	3.92
160	0.25	4.753(−4)	10.8	3.43	7.301(−5)	15.5	3.95
320	0.25	5.400(−5)	8.80	3.13	3.757(−6)	19.4	4.28
640	0.25	4.644(−6)	11.6	3.53	2.801(−7)	13.4	3.74

show that the new cut-cell method can achieve uniformly fourth order accuracy in the entire irregular domain. In this case, the grid stencil is chosen to be $q = 6$ for computing derivatives for the irregular points. Due to the increase of the order of the schemes, the classification of the grid points is:

$$\begin{cases} i = 1, \ldots, (n-4) & \text{regular points} \\ i = n-3, n-2, n-1 & \text{irregular points} \end{cases} \tag{46}$$

The fourth-order schemes for the three irregular points can be written as

$$\left(T_x^I\right)_{n-3} = \frac{a_{4,6}T_{n-5} + a_{4,5}T_{n-4} + a_{4,4}T^I_{n-3} + a_{4,3}T^I_{n-2} + a_{4,2}T^I_{n-1} + a_{4,1}T^B}{\Delta h} \tag{47}$$

$$\left(T_x^I\right)_{n-2} = \frac{a_{3,5}T_{n-4} + a_{3,4}T^I_{n-3} + a_{3,3}T^I_{n-2} + a_{3,2}T^I_{n-1} + a_{3,1}T^B}{\Delta h} \tag{48}$$

$$\left(T_x^I\right)_{n-1} = \frac{a_{2,5}T_{n-4} + a_{2,4}T^I_{n-3} + a_{2,3}T^I_{n-2} + a_{2,2}T^I_{n-1} + a_{2,1}T^B}{\Delta h} \tag{49}$$

where T^I represents the irregular points and T^B the boundary point. The corresponding coefficients $a_{i,j}$ can be found in Table A3.

Table 2 shows the numerical errors and order of accuracy, based on both the L^∞ and L^2 norms, for the uniform fourth-order cut-cell method. Compared with the results of the local third-order boundary closure schemes listed in Table 1, the numerical errors are reduced by an half by using the current fourth-order boundary closure. Table 2 also shows that as the number of grid points N increases, the computational order of accuracy approaches the expected value of 4 in both L^∞ and L^2 norms. These results validate our cut-cell method in the sense that it can achieve uniformly high-order accuracy for problems with irregular domains.

In our cut-cell method, the critical value Θ is used to decide whether a grid point can be classified as a dropped point or not. This is a necessary measure in order to avoid the small cell problem in a cut-cell scheme. The best value of Θ may depend on the specific problem. Generally speaking, extra computational errors may be introduced by the boundary scheme when Θ exceeds 1. On the other hand, the time step is restricted by the small Θ for explicit temporal discretization. In order to release the time step restriction and minimize the computational errors of the boundary numerical scheme, the suggested range of this parameter is $\Theta \in [0.5, 1]$.

In order to investigate the effect of the value of Θ on the accuracy of this test case, several cases with different Θ values, ranging from 0.25 to 1.5, are computationally tested. The results are listed in Table 3. When the ratio is reduced below 0.25, none of points in boundary stencil is classified as dropped point, thus all of the computational results are the identical. As shown in Table 3, the computational error $\|E_N\|_\infty$ (the accuracy of the problem) always decreases as the Θ decreases. Table 3 also shows that p is close to 4 for these cases. With $\Theta = 0.5$, the computational order of accuracy for our cut-cell method is the highest among all the test cases computed from grids sets $N = 80$ and $N = 160$. But the small differences of the actual value of p do not indicate the level of accuracy among these cases. The bottom line is that we need to pick a reasonable value of Θ to avoid small cell problem and Table 3 shows that the accuracy is approximately maintained.

Table 2
Numerical errors of the fourth-order cut-cell method with fourth-order boundary closure schemes for computing the one-dimensional boundary value problem with an irregular domain.

N	Θ	$\|E_N\|_\infty$	Ratio	p	$\|E_N\|_2$	Ratio	p
20	0.5	1.451(0)			8.709(−1)		
40	0.5	3.001(−2)	48.4	5.59	1.691(−2)	45.6	5.51
80	0.5	1.976(−3)	15.2	3.92	1.036(−3)	16.3	4.02
160	0.5	1.112(−4)	17.8	4.15	6.231(−5)	15.3	3.94
320	0.5	9.051(−6)	15.6	3.97	3.684(−6)	19.6	4.23
640	0.5	6.160(−7)	14.6	3.87	2.372(−7)	15.6	3.96

Table 3
Computational order of accuracy for the fourth-order cut-cell method with fourth-order boundary closure scheme with different values of Θ.

Θ	80 Grids $\|E_N\|_\infty$	160 Grids $\|E_N\|_\infty$	Ratio	p
2.0	2.201(−3)	1.947(−4)	11.3	3.50
1.0	1.976(−3)	1.419(−4)	13.9	3.79
0.6	1.976(−3)	1.419(−4)	13.9	3.79
0.5	1.976(−3)	1.112(−4)	17.8	4.15
0.25	1.376(−3)	1.112(−4)	12.3	3.62

4.2. One-dimensional hyperbolic equation in an irregular domain

As the second test case of our new high-order cut-cell method, we consider one-dimensional hyperbolic equation:

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0, \quad x \in [0, C_\delta] \tag{50}$$

Equation is defined in an irregular domain $\Omega = [0, C_\delta]$ with one side boundary condition imposed as:

$$u(C_\delta, t) = \sin\left\{\frac{\omega\pi(ct - C_\delta)}{c}\right\}, \quad t \geqslant 0$$

where C_δ is a constant smaller than 1.0. The set of uniform N grid points are used to cover the regular computational domain $\Omega' = [0,1]$ as in Section 4.1. The specific parameters used in our computational test are:

$$\begin{cases} c = -1.0 \\ \omega = 5.0 \\ C_\delta = \sqrt{3}/2 \end{cases}$$

The exact solution can be written as,

$$u_e(x,t) = \sin\{5\pi(t+x)\}, \quad t \geqslant 0,\ x \in [0, \sqrt{3}/2] \tag{51}$$

The stability of explicit finite discretization of equation in irregular domain is restricted by the small grids spacing θ in Fig. 7, e.g., in this one-dimensional case, the length of time step of explicit schemes is restricted by $O\{\min(\Delta h, \theta)\}$. Thus the explicit discretization may be computationally expensive if the non-uniform grid spaceing θ is too small. To prevent potential stringent time step restriction, a second order Crank–Nicholson method is used for temporal discretization. Crank–Nicholson is unconditional stable and simple in implementation. We tested our cut cell method up to fourth-order accuracy in previous boundary value problem. For hyperbolic equation, to achieve uniformly fourth-order accuracy in both temporal and spatial approximation, the time step Δt is chosen such that

$$\Delta t = c(\Delta h)^2 \tag{52}$$

A set of uniform M grids is generated for temporal discretization with spacing Δt. The grids generation and definition in spatial is the same as those in Section 4.1. The Crank–Nicholson scheme can be written as:

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} = \frac{1}{2}\left\{(u_x)_i^{j+1} + (u_x)_i^j\right\}, \quad i = 1, \ldots, n-1,\ j = 1, \ldots, m-1 \tag{53}$$

where $(u_x)_i^j$ represent the spatial approximation of first order derivatives of u at time t^j and grids i.

The third- and fourth-order cut-cell spatial discretization for $(u_x)_i^j$ as in Eqs. (41)–(43) and Eqs. (47)–(49) can be applied into Eq. (53). The numerical value at boundary $i = 0$ is computed by using fourth-order one-side finite-difference scheme as,

$$(u_x)_0 = \frac{4u_1 - 6u_2 + 4u_3 + u_4}{\Delta h} \tag{54}$$

At each time t^j the implicit numerical solutions are obtained by using iteration method to solve the linear system.

Fig. 9 compares the exact solution and the solution by using the uniform fourth-order cut-cell method in Eqs. (47)–(49) and Crank–Nicholson method with grids $N = 320$ at $t = 1.0$ (s). A good agreement is achieved between the two sets of solutions. Table 4 gives the computational errors, in L^∞ and L^2 norms. As the number of grid points N increases, fourth-order accuracy is obtained uniformly. These results validate our cut-cell method in the sense that it can achieve uniformly high-order accuracy for hyperbolic equation with irregular domains.

4.3. Two-dimensional elliptic equation in an irregular domain

The third test case for the current high-order two-dimensional cut-cell method is the same case used by Leveque and Li [36] to test their immersed interface method. Specifically, it is an elliptic equation in an irregular domain given by

$$u_{xx} + u_{yy} = \int_\Gamma 2\delta(x - X(s))\delta(y - Y(s))ds \tag{55}$$

where the interface Γ is a circle defined by $x^2 + y^2 = 1/4$. The computational domain is $(x,y) \in [-1,1]^2$ as shown in Fig. 9. The Dirichlet boundary condition is specified along the boundary with the exact solution:

$$u(x,y) = \begin{cases} 1 & \text{if } r \leqslant 1/2 \\ 1 + \log(2r) & \text{if } r > 1/2 \end{cases} \tag{56}$$

The uniform Cartesian grids of $N \times N$ grid points shown in Fig. 10 are used for the problem. Boundary points are created by the intersection of the gridlines with the interface. The boundary points and irregular points in cut-cell grids are defined in

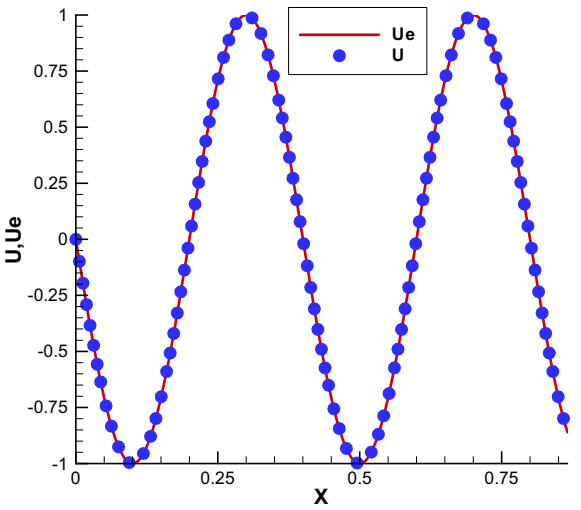

Fig. 9. Comparison of numerical solution $U(x)$ computed by the fourth-order cut-cell method and the exact solution $U_e(x)$ ($N = 320$) at $t = 1.0$ (s).

Table 4
Numerical errors of the fourth-order cut-cell method with fourth-order boundary closure schemes for computing the one-dimensional hyperbolic equation with an irregular domain at $t = 0.005$ (s).

N	Θ	$\|E_N\|_\infty$	Ratio	p	$\|E_N\|_2$	Ratio	p
20	0.5	1.890(0)			1.209(0)		
40	0.5	5.887(−2)	32.1	5.00	3.311(−1)	36.5	5.51
80	0.5	4.912(−3)	12.0	3.58	2.212(−3)	15.0	4.02
160	0.5	4.012(−4)	12.2	3.61	1.456(−4)	15.2	3.93
320	0.5	2.902(−5)	13.8	3.80	8.001(−6)	18.2	4.19

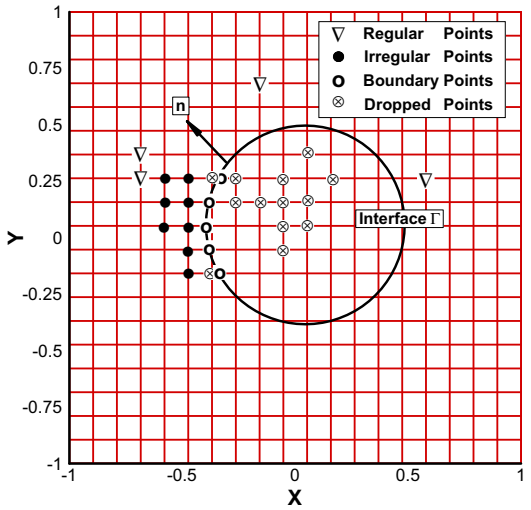

Fig. 10. Two-dimensional computational domain and grid for the current cut-cell method.

Section 3. A critical distance Θ is used as a criterion for the determination of dropping points. Consequently, all four types of grid points, which are regular, irregular, dropped and boundary points, are defined for the entire computational domain.

For the regular points, we extend the fourth-order schemes given by Eq. (41) for the one-dimensional case to the derivatives along the x and y directions. For example, for the derivative in the x direction, we have:

$$
\begin{aligned}
(u_x)_{i,j} &= \frac{5u_{i-2,j} - 40u_{i-1,j} + 40Tu_{i+1,j} - 5u_{i+2,j}}{60\Delta h} \quad \text{(for regular points inside the domain)} \\
(u_x)_{i=1} &= \frac{-3u_{0,j} - 10u_{1,j} + 16u_{2,j} - 6u_{3,j} + u_{4,j}}{12\Delta h} \\
(u_x)_{i=N-1} &= -\frac{-3u_{N,j} - 10u_{N-1,j} + 16u_{N-2,j} - 6u_{N-3,j} + u_{N-4,j}}{12\Delta h}
\end{aligned} \tag{57}
$$

Similar formulas can be obtained for the derivatives in the y direction.

For irregular points, we extend the non-uniform-grid finite difference schemes for the one-dimensional problem to the current two-dimensional problem in a dimension by dimension manner. For the case of third-order schemes for the irregular points, the formulas of Eqs. (42) and (43) are extended to derivatives in the x and y directions, respectively. The treatments for the dropped points and the boundary points have been described in Section 3. In order to obtain fourth-order global accuracy of method, we use a fourth-order polynomial interpolation for the valuables $u_{i,j}$ of the dropped points.

In this paper, we have tested the following two versions of the cut-cell methods with different orders of accuracy for the boundary closure schemes for the irregular points, while both versions use the same fourth-order schemes for the regular points:

- Version 1: Second-order cut-cell boundary closure schemes with coefficients given by Table A1. Table A1 are used for the irregular points. This version is expected to be globally third-order accurate.
- Version 2: Third-order cut-cell s boundary closure schemes with coefficients given by Table A2. Table A2 are used for the irregular points. This version is expected to be globally fourth-order accurate.

Fig. 11 compares the numerical solution computed by Version 2 of the fourth-order cut-cell method for the two-dimensional example and the exact solution for the case of 80×80 grids. The value of Θ is 1.0 for this case. This figure shows a very good agreement between the exact and numerical solutions. There is no spurious oscillation for the numerical solutions in the region adjacent to the interface.

Tables 5 and 6 show the computational errors, in L^∞ and L^2 norms respectively, with four different grid sets. Table 6 shows that solutions of Version 2 of the cut-cell method approach third and fourth orders in L^2 norm as the values of N increase. This is consistent with the global fourth-order accuracy of the scheme. As expected, Table 5 shows that the orders in infinity norm are lower than those of the second norm. Overall, the fourth-order cut-cell method with third-order boundary closure schemes produces a fourth-order global accuracy in L^2 norm for the solutions to the two-dimensional problem.

4.4. Two-dimensional hyperbolic equation in an irregular domain

The fourth case for testing the current high-order cut-cell method is the two-dimensional convection equation:

$$
\frac{\partial u}{\partial t} + c_1\frac{\partial u}{\partial x} + c_2\frac{\partial u}{\partial y} = 0 \quad \text{for } x, y \in [-1,1]^2 \cap \Omega^+ \tag{58}
$$

where the domain Ω^+ is defined as $\Omega^+ = \{x,y: (x-1)^2 + (y-1)^2 \leqslant 2\}$ as shown in Fig. 12. The specific parameters used in our computation are:

$$
\begin{cases} c = 1.0 \\ \omega = 5.0 \end{cases}
$$

The time relevant boundary condition is specified along the interface $\Gamma = \{x,y: (x-1)^2 + (y-1)^2 = 2\}$ within the domain. The exact solution is:

$$
u(x,y,t) = \frac{\sqrt{2}}{2}\sin\{5(x-t)\} + \frac{\sqrt{2}}{2}\sin\{5(y-t)\} \quad \text{for } x, y \in [-1,1]^2 \cap \Omega^+ \tag{59}
$$

Uniform M grid points are generated for temporal discretization with spacing Δt, and $N \times N$ Cartesian grid points are generated for spatial discretization with spacing Δh in this irregular domain. The Crank–Nicholson scheme in Eq. (53) is used for temporal approximation. Δt is calculated as in Eq. (52).

Similar to one-dimensional hyperbolic problem in Section 4.2, the high-order two-dimensional cut-cell discretization in Section 4.3 for $(u_x)_i^j$ and $(u_y)_i^j$ can be applied to discretize the spatial derivatives. The outlet boundary condition is obtained by using one-side finite difference scheme as in Eq. (52).

Fig. 12(b) shows numerical solution computed by Version 2 of the fourth-order cut-cell method in Section 4.3 and Crank–Nicholson method. Table 7 shows the computational errors, in L^∞ and L^2 norms respectively, with four different grid sets. The computational result demonstrates that current high-order cut-cell method can obtain fourth-order accuracy for two-dimensional hyperbolic equation.

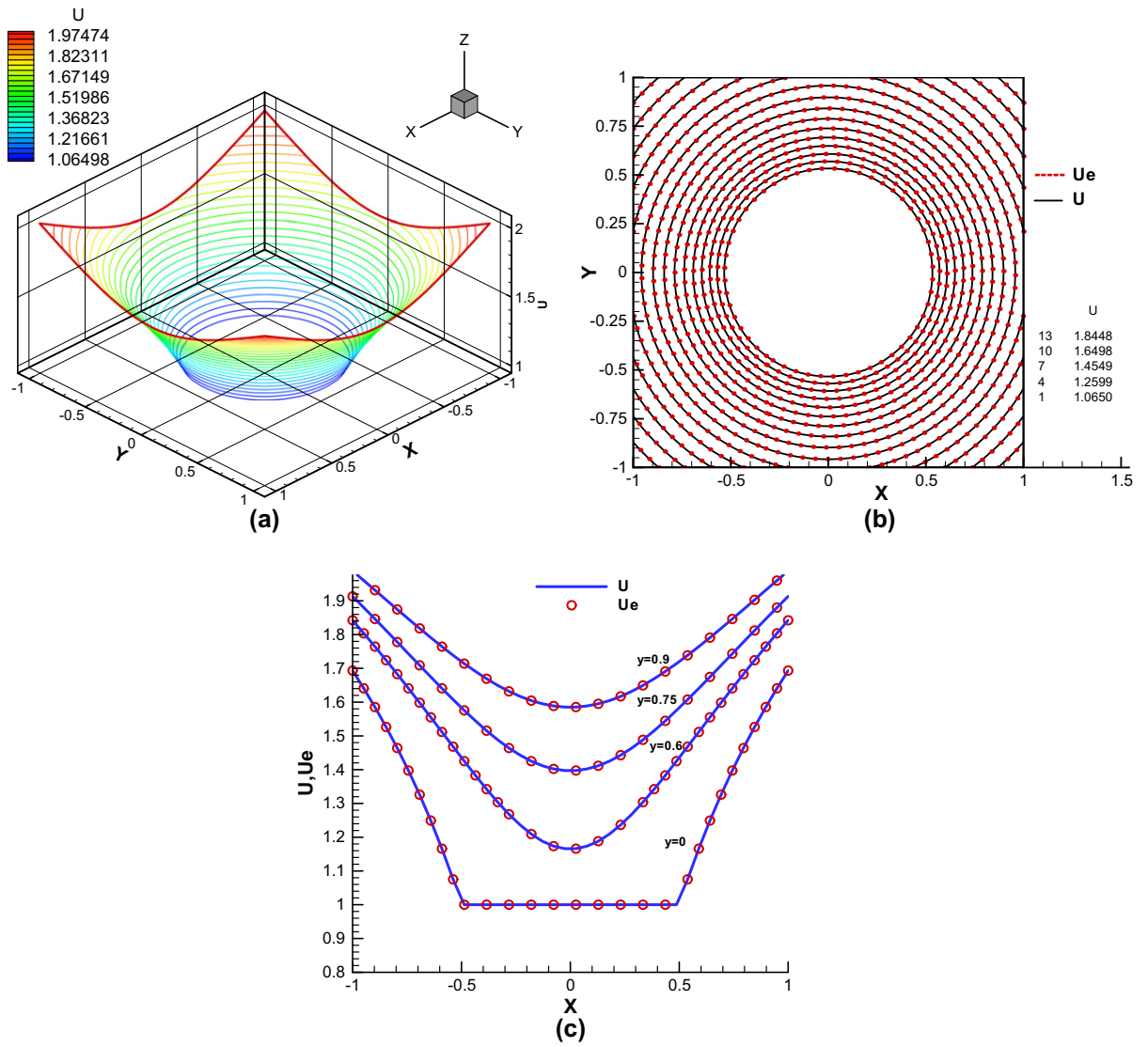

Fig. 11. Comparison of the solution, $u(x,y)$, computed by Version 2 of the fourth-order cut-cell method ($O(h^4)$) for the two-dimensional example and the exact (u_e) (grid: 80 × 80): (a) and (b) are contours of the solution, (c) is distribution along the x direction.

Table 5
Computational errors in L^∞ norm of the two versions of the cut-cell methods ($O(h^3)$ and $O(h^4)$) for the two-dimensional test case of Leveque and Li [36].

N	Cut-cell method 1				Cut-cell method 2			
	Θ	$\|E_N\|_\infty$	Ratio	p	Θ	$\|E_N\|_\infty$	Ratio	p
20	0.5	1.688(−3)			1.0	7.075(−4)		
40	0.5	1.528(−4)	11.0	3.46	1.0	6.285(−5)	11.3	3.50
80	0.5	1.714(−5)	8.92	3.16	1.0	7.295(−6)	8.61	3.10
160	0.5	2.217(−6)	7.73	2.95	1.0	6.337(−7)	11.5	3.53

4.5. Two-dimensional hypersonic viscous flow over a flat plat with surface roughness

Having tested the accuracy of the current high-order cut-cell method for linear model equations, the new method is used to compute hypersonic viscous flow over a flat plate with an isolated surface roughness element (Fig. 2). Both steady and unsteady flows are considered. The freestream flow conditions are the same as those used in Maslov's experiment [51] as follows,

Table 6
Computational errors in L^2 norm of the two versions of the cut-cell methods ($O(h^3)$ and $O(h^4)$) for the two-dimensional test case of Leveque and Li [36].

N	Cut-cell method 1				Cut-cell method 2			
	Θ	$\|\|E_N\|\|_2$	Ratio	p	Θ	$\|\|E_N\|\|_2$	Ratio	p
20	0.5	4.791(−4)			1.0	1.967(−4)		
40	0.5	4.040(−5)	11.9	3.56	1.0	1.437(−5)	13.7	3.77
80	0.5	2.919(−6)	9.54	3.25	1.0	9.049(−7)	15.9	3.99
160	0.5	2.945(−7)	9.91	3.30	1.0	5.773(−8)	15.7	3.97

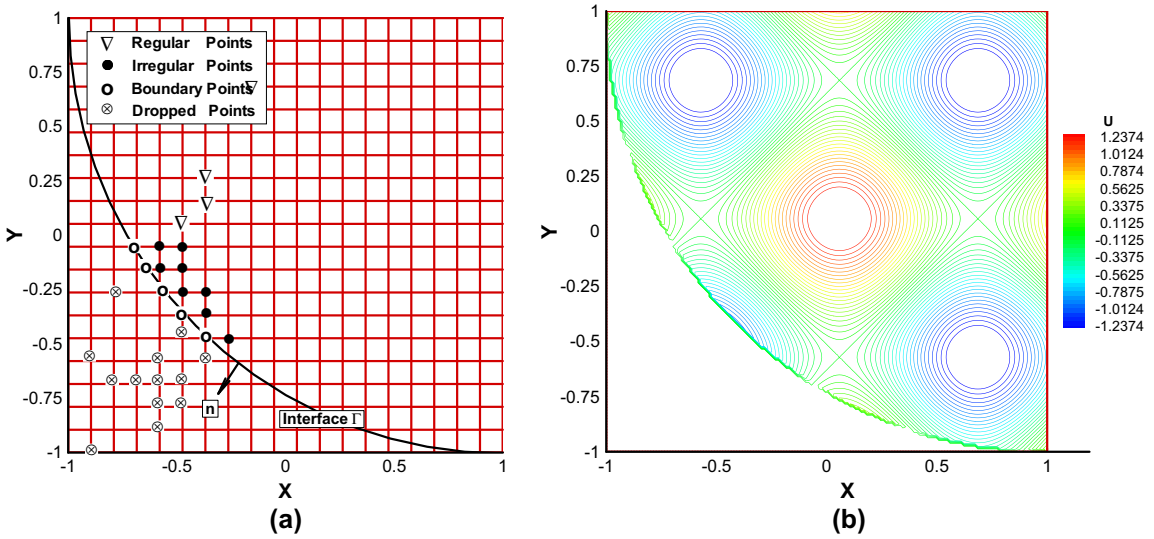

Fig. 12. (a) Two-dimensional computational domain and grids for the current cut-cell method, (b) computational solution, $u(x,y)$, computed by fourth-order cut-cell method ($O(h^4)$) for the two-dimensional unsteady (grid: 80×80, $t = 1$ s).

Table 7
Computational errors in L^2 norm of the two versions of the cut-cell methods ($O(h^3)$ and $O(h^4)$) for the two-dimensional convection equation.

N	Cut-cell method 1				Cut-cell method 2			
	Θ	$\|\|E_N\|\|_2$	Ratio	p	Θ	$\|\|E_N\|\|_2$	Ratio	p
20	0.5	7.213(−4)			1.0	2.453(−4)		
40	0.5	4.127(−5)	17.5	4.13	1.0	1.422(−5)	17.3	4.11
80	0.5	2.831(−6)	14.1	3.82	1.0	8.153(−7)	16.9	4.07
160	0.5	2.808(−7)	9.99	3.33	1.0	5.411(−8)	15.1	3.91

$$\begin{cases} P_r = 0.72, \quad R_\infty = \rho_\infty * u_\infty/\mu_\infty = 1.32 \times 10^6/\text{m} \\ M_\infty = 5.92, \quad T_\infty = 48.69\ \text{K}, \quad P_\infty = 742.76\ \text{Pa} \end{cases} \tag{60}$$

where M_∞, T_∞, P_∞, P_r, R_∞ are Mach number, temperature, pressure, Prantle number and unit Reynolds number, respectively. The flat plate is assumed to be isothermal with a constant temperature of $T_w = 350.0$ K. The total length of the flat plate is about 1.69 m.

The steady mean flow solutions are calculated by using the fifth-order shock-fitting method discussed in Section 3. But in the leading edge region of the flat plate, there is a singularity at the tip of the plate and the high-order shock fitting method cannot be used there. Thus a second-order TVD shock-capturing method is employed to calculate a small local flow field around the tip of the flat plate. The computational domain for the TVD calculations starts at $x = -0.006$ m and ends at a very short distance downstream of the leading edge at $x = 0.003$ m. A total 241×121 grid points are used. A TVD scheme, which follows that used by Lee et al. [52], is applied to Eq. (12). The semi-discrete system of ordinary differential equations are then solved by using a fourth-order Runge–Kutta method.

Having obtained the steady state solutions at the leading edge, we then use the solution of the TVD scheme as the inlet condition to start the subsequent shock-fitting calculations. The computational domain for the high-order shock-fitting methods starts at $x = 0.003$ m and ends at $x = 1.68784$ m. In actual simulations, the computational domain is divided into

30 zones, with total of 5936 grid points in the streamwise direction and 121 points in the wall-normal direction. As mentioned above, the second zone uses the results of the first zone of the second-order TVD solution as the inlet condition. A later zone uses the interpolation of its former zone's data as the inlet condition.

In this section, unless stated otherwise, most flow variables are presented as dimensionless ones. The main exception is that the x and y coordinates are presented in dimensional form so that the simulational results can be easily related to the experimental setup. For other variables, we nondimensionalize the flow velocities by the freestream velocity u_∞, density by ρ_∞, pressure by $\rho_\infty(u_\infty)^2$, and temperature by T_∞.

4.5.1. Steady flow solution without surface roughness

The solutions for the steady viscous flow over the flat plate with no surface roughness are first obtained. Fig. 13 shows the streamwise velocity and temperature profiles along the wall-normal direction at $x = 0.1676$ m near the wall. The current numerical solutions obtained by the fifth-order shock-fitting scheme are compared with the self-similar boundary layer solution. In order to compare with the self-similar solution of the boundary layer, the y coordinate is nondimensionalized by $\sqrt{x\mu_\infty/\rho_\infty u_\infty}$. Fig. 13 illustrates that the results of the current numerical simulation agree very well with the theoretical solutions near the wall. The second-order TVD scheme is accurate enough to be used as the inlet condition of the fifth-order shock-fitting method downstream of the leading edge. But in region at $x = 0.1676$ m near the bow shock, the two solutions are different since the self-similar solution does not consider the shock effect. For boundary-layer instability simulation, the region near the wall is the most important. However, the shock is also important in receptivity problems where the instability waves are excited in the freestream, and enter the boundary layer via the bow shock. In this case, it is necessary to consider the bow shock and its interaction with wave fields.

4.5.2. Steady flow solution with an isolated surface roughness

An isolated roughness element of smooth shape is placed on the surface of the flat plate at $x = 0.185$ m downstream. The shape of the surface roughness is chosen to be a two-dimensional bump, governed by the following elliptic equation,

$$\frac{(x-x_c)^2}{a^2}+\frac{y^2}{b^2}=h^2 \tag{61}$$

The computations of this study are performed under the following parameters: $a = 2$, $b = 1$, $x_c = 0.185$ m and $h = \delta/2$ where $\delta = 0.00081$ m which corresponds to the boundary layer thickness at x_c as shown in Fig. 14.

In this paper, a third-order cut-cell method described in Section 3 is used to compute the two-dimensional viscous hypersonic flow over the flat plate with the roughness element. As described in Section 3, a coordinate transformation is employed to transform the physical domain shown in Fig. 14 into a rectangle computational domain with a set of Cartesian grid. The optimal transformation formula is determined by the specific physical problem considered. For viscous flow over a flat plate, it is necessary to cluster more grid points at the bottom wall surface in order to resolve the viscous boundary layer. In this paper, a two-step mapping procedure is used to obtain better resolution inside the viscous boundary layer.

In the first step of the transformation, the entire physical domain shown in Fig. 14 is transformed into a square domain defined on $[0,1] \times [0,1]$ as an intermediate coordinate space. The transformation relation is defined as:

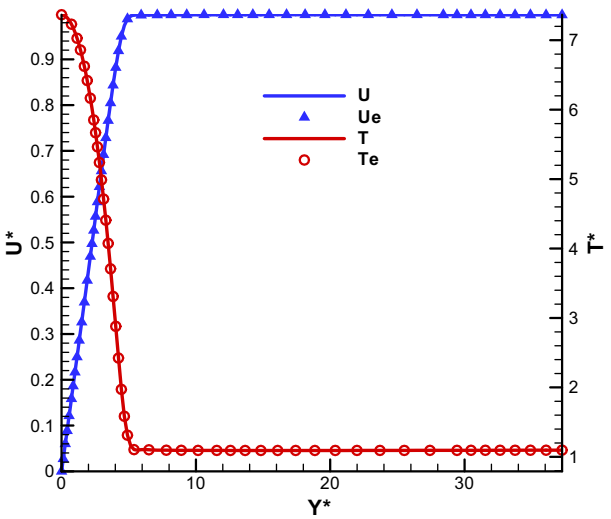

Fig. 13. Comparison of the current shock-fitting results with theoretical solutions (T_e and u_e) for temperature and wall normal velocity distributions long a wall-normal grid line located at $x = 0.1676$ m on the plate surface.

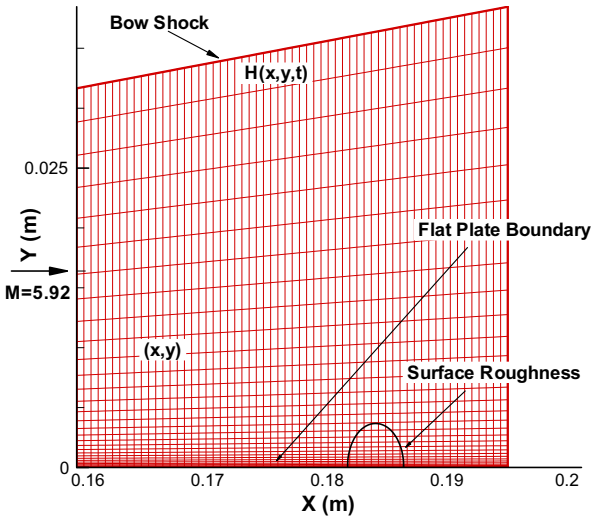

Fig. 14. Computational grid for hypersonic flow over a flat plate with an isolated surface roughness. The grid size is 241 × 121 (showing one out of four grid points in both directions).

$$\begin{cases} X = \frac{x - x_{strart}}{L} \\ Y = \frac{y}{H(x,y)} \end{cases} \tag{62}$$

where (X, Y) is defined under the intermediate coordinate system, L is the streamwise length of the flat plate in physical domain, x_{start} is the streamwise coordinate of starting point of computation in each zone, and the distance along η direction between the solid wall and bow shock is referred to H.

In the second step of the transformation, the intermediate plane in the (X, Y) space is mapped into the final computational domain (ξ, η) in order to cluster more grid points into the viscous boundary layer near the flat plate surface. In the present study, an exponential stretch function is used to cluster grids in the η direction as follows,

$$\begin{cases} X = \xi \\ Y = -\frac{\left(\frac{\beta+1}{\beta-1}\right)^{(1-\eta)} - 1}{\left(\frac{\beta+1}{\beta-1}\right)^{(1-\eta)} + 1}\beta \end{cases} \tag{63}$$

where β is the stretching parameter. The value of $\beta = 1.01$ is used in this paper. With this β value, about 50% of the total grid points in the η direction are clustered inside the boundary layer. The combination of the transformations of these two steps leads to overall transformation given by Eq. (11).

In the computational domain (ξ, η), the baseline grid is a set of Cartesian grid similar to the schematic shown in Fig. 2(b). The interaction of the roughness surface with the grid lines creates cut cells, which are treated by the current cut-cell method. The surface equation (61) of the roughness surface is transformed into the computational domain in order to apply the cut-cell method. By substituting Eqs. (62) and (63) into the roughness surface equation (61), the analytical equation for roughness surface in the computational domain can be written as

$$\begin{aligned} & f(\xi, \eta) = \frac{(L\xi)^2}{a^2} + \frac{\ell H(\xi L + x_{start})^2}{b^2} - h^2 = 0 \\ & \text{where} \quad \ell = \frac{\left(\frac{\beta+1}{\beta-1}\right)^{(1-\eta)} - 1}{\left(\frac{\beta+1}{\beta-1}\right)^{(1-\eta)} + 1}\beta \end{aligned} \tag{64}$$

From Eq. (64), we can calculate the coordinates of the boundary points by computing the coordinates of the intersection between the roughness surface and the grid lines. For a given value of ξ along the roughness surface, the Newton iteration method is employed to solve Eq. (64) for the corresponding value of η on the surface for a boundary point, and vice versa. Having computed the boundary points, we can identify all irregular and dropped points near the roughness surface. The remaining grid points are regular points.

After computing all coordinates of the irregular, regular, boundary and dropped points as shown in Fig. 4, the current high-order cut-cell method is applied to discretize Eq. (12). The bow shock is treated by a high-order shock-fitting method. The overall accuracy of simulation is determined by the order of scheme used for computing the inviscid and viscous flux terms. In this paper, the results of a third-order ($p = 3$) cut-cell method is presented. In simulation, the numerical dissipation

coefficient α is chosen within range [0.25–6.0]. All the simulations are very stable. Thus the numerical dissipation of the interior scheme may only have minor influence on the overall stability.

Fig. 15 shows steady flow solutions in the region surrounding the roughness element computed by the third-order cut-cell method, in both the streamline pattern and the contours of wall normal velocity components. In order to assess the numerical accuracy of the solutions, steady solutions are obtained by using the following two sets of grids: 241×121 and 441×241. The grid refinement study for $x \in (0.159, 0.195]$ m is conducted both in x and y directions. The number of grids points in y direction is doubled to 241 where the value in the inlet is obtained by using sixth-order interpolation from the coarse grids. The total grid points for the refine-grid case are 441×241. The wall-normal velocities computed by fine and coarse grids are plotted with the same levels of contour line in Fig. 15(b). The good agreement between these two sets of results suggests that the grid resolution used in the current simulations is high enough for the steady flow simulation. By integrating the local velocity vectors, the streamline patterns are obtained in Fig. 15(a), which shows that the flow contains two separation regions as it approaches the roughness. The two separation regions are located immediately before and after the roughness element, respectively. The flow speed in the separation region after the roughness is slower than that for the one before the roughness. The streamline patterns also show that the local flow around the roughness does not form parallel flow patterns used in Linear Stability Theory.

Fig. 16 shows the steady-flow pressure contours for flow over a flat plate with an isolated roughness computed by the third-order cut-cell method. The roughness element, which centers at $x_c = 0.185$ m, can be seen on the lower surface in the figure. Because the flow is supersonic behind the bow shock, a family of Mach waves is generated by flow over the roughness. The compression waves are followed by expansion waves when the flow expands around the roughness surface. These Mach waves are approximately parallel to the bow shock interface in the later zones.

The steady flow solutions demonstrate that, affected by the roughness, hypersonic boundary layer over the flat plate is modified significantly in the regions both downstream and upstream of the roughness element. The parallel flow assumption is no longer valid for flow near the roughness. Consequently, normal-mode linear stability analysis may not be accurate in this flow region. Thus a different boundary layer instability mechanism may be introduced. On the other hand, the effects of the roughness is most significant in the regions surrounding the roughness. The roughness effects on the steady flow decay if we move further downstream. Thus the flow is approximately parallel again in the far downstream of the surface roughness. Linear Stability Theory (LST) can be applied to analyze the flow field which is approximately parallel. Good agreement between numerical and theoretical solutions can be achieved. More details can be found in [53].

4.5.3. Receptivity of hypersonic flow with surface roughness to wall blowing and suction

In this section, the new third-order cut-cell method is applied to the computation of transient responses of the same Mach 5.92 boundary layer to forcing waves introduced by a blowing and suction slot located on the plate surface. It is termed the receptivity problem of the boundary layer to forcing waves [5]. The receptivity problem is critical to the understanding of physical mechanisms of hypersonic boundary layer transition. The receptivity of the same hypersonic boundary layer to wall blowing and suction has been studied by Wang and Zhong [54,55] for smooth surface without roughness. In this paper, we study the additional effects of the isolated surface roughness on the receptivity process by using the current third-order cut-cell method. The current results are compared with those of [54,55] to study the effects of surface roughness on the receptivity process.

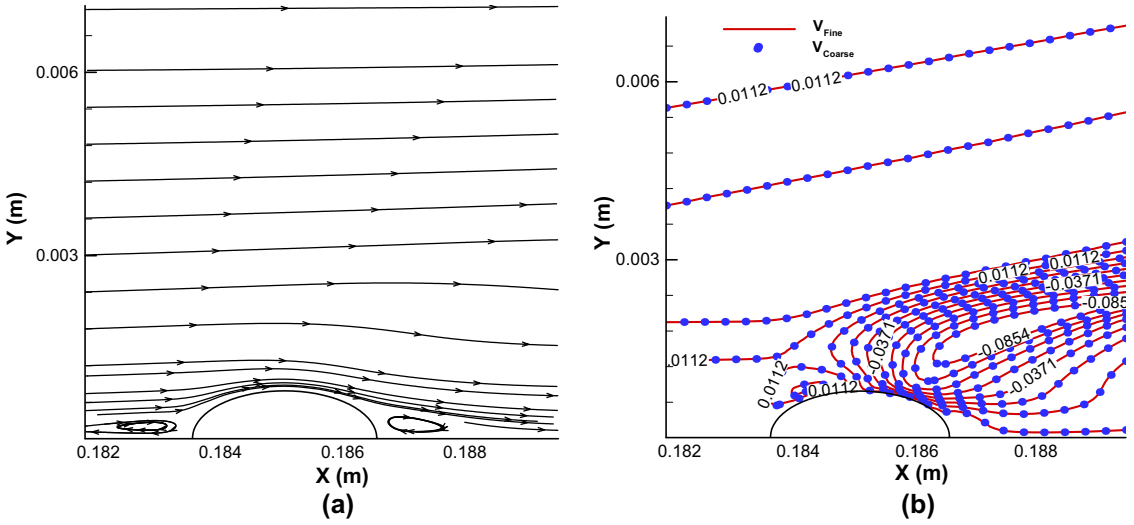

Fig. 15. Hypersonic flow solutions in the region near the roughness element computed by the third-order cut-cell method: (a) streamline pattern, (b) contours of wall normal velocity components computed by using two sets of grids (coarse grid: 241×121, fine grids: 441×241).

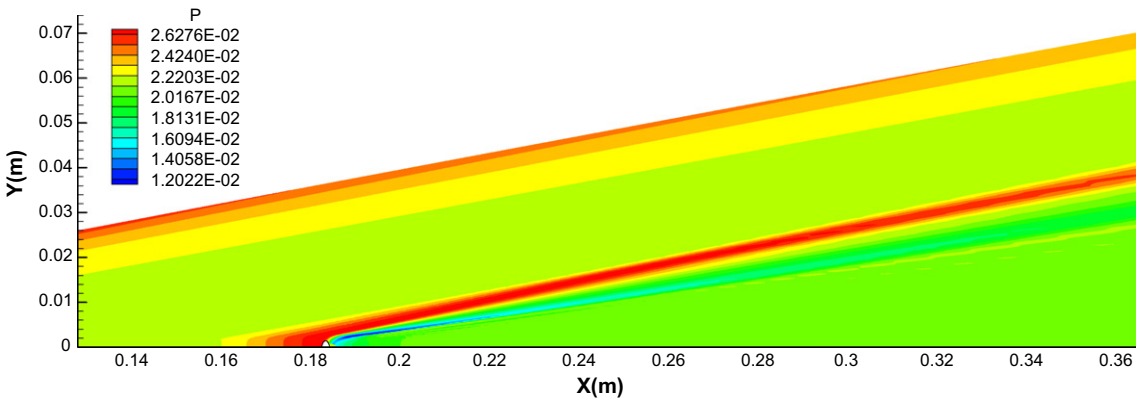

Fig. 16. Pressure contours for two-dimensional hypersonic flow over a flat plate with an isolated roughness computed by the third-order cut-cell method. The total grid points in this range of the flow field is 1920×121.

The receptivity study is mainly concerned with the excitation of instability waves, the characteristics of which can be analyzed by the linear stability theory [56]. The instability theory analyzes the propagation of individual sinusoidal waves in the streamwise direction inside the boundary layer. These waves are referred as Tollmien–Schlichting (T–S) waves for low speed flow. The instability waves are vorticity waves, whose amplitudes vary though the boundary layer and die off exponentially outside the boundary layer. For the case of small perturbations in the flow field, the perturbations of flow variables can be written in the form of a normal mode, i.e.,

$$\begin{Bmatrix} u \\ v \\ p \\ \theta \end{Bmatrix} = \begin{Bmatrix} \tilde{u}(y) \\ \tilde{v}(y) \\ \tilde{p}(y) \\ \tilde{\theta}(y) \end{Bmatrix} e^{i(\int \alpha\, dx - \omega t)} \tag{65}$$

where θ is the perturbation of temperature. The frequency of the waves is ω and the streamwise wave number is represented by α.

Substituting Eq. (65) into a linearized version of the full Navier–Stokes equation (1), we obtain a system of linearized equations of the stability theory. These equations are required to satisfy a number of boundary conditions. By imposing the homogenous physical conditions, the number of solutions of an eigen problem to the linearized equations with specific value of α and ω is constrained. The relation for the instability wave parameter α and ω are referred as dispersion relations in the following form,

$$\omega = \omega(\alpha) \tag{66}$$

Extensive numerical and theoretical research has been conducted to solve the linearized Navier–Stokes equations and many characteristics regarding the instability waves in hypersonic boundary layers have been discovered [51,56–59]. Mack [56] identified the unstable modes by using the linear stability theory. He showed that inside a supersonic boundary layer, there are multiple higher instability modes in addition to the first mode, which is the compressible counterpart of T–S waves in the incompressible boundary layers. These instability modes in the supersonic boundary layer are termed as first mode, second mode, third mode, etc. The second mode is also called the Mack mode. For supersonic boundary layer with Mach number larger than four, the second Mack mode is the most unstable mode, and it plays an important role in hypersonic boundary layer transition.

To excite the propagation of small disturbances inside the boundary layer, a blowing and suction slot is imposed as periodic-in-time boundary conditions for the perturbations of the mass flux on the wall. The blow-suction slot is located at $x = 0.030$ m and spreads over several grids spaces. The perturbations in the blowing and suction slot are governed by the following function:

$$\rho v = q_0 g(l) \sum_{n=1}^{15} \sin(w_n t) \tag{67}$$

where q_0 is an amplitude parameter, $w_n = nf_1$ is circular frequency of this multi-frequency perturbation, and $g(l)$ is a non-dimensional x-direction profile function defined as

$$g(l) = \begin{cases} 20.24l^5 - 35.4375l^4 + 15.1875l^3 & (l < 1) \\ -20.24(2-l)^5 + 35.4375(2-l)^4 - 15.1875(2-l)^3 & (l \geqslant 1) \end{cases} \tag{68}$$

The variable l is the non-dimensional parameter associated with current coordinate of this blowing and suction,

$$l(x) = \frac{2(x - x_i)}{(x_e - x_i)} \tag{69}$$

The basic frequency f_1 is given by $f_1 = 50$ kHz, and the other 14 different frequencies are imposed as $f = 100, 150, 200, \ldots$ kHz for $n = 2, 3, 4, \ldots$ A Fast Fourier Transformation (FFT) technique is used to separate the results with different frequencies. This steady and unsteady flow conditions used in the current simulation are the same as those used in the simulation conducted by Wang and Zhong [54] for the cases of smooth wall without roughness. Balakumar [40] also investigated the receptivity of a 2-D roughness to acoustic waves and found the isolated roughness does not contribute much in generating unstable disturbances.

Fig. 17 shows the contours of pressure perturbations for cases with surface roughness and with combined 15 forcing frequencies. The roughness is located at $x_c = 0.185$ m. The pressure disturbances at flat plate surface are reduced after the instability waves pass the surface of roughness element.

Fig. 18 compares the amplitudes of surface pressure perturbations for cases with roughness and without roughness at two different frequencies of $f = 100$ kHz and $f = 150$ kHz. For the case without roughness, the wave amplitudes grow exponentially for $f = 100$ kHz, which is an indication of the second-mode instability in the boundary layer. On the other hand, for

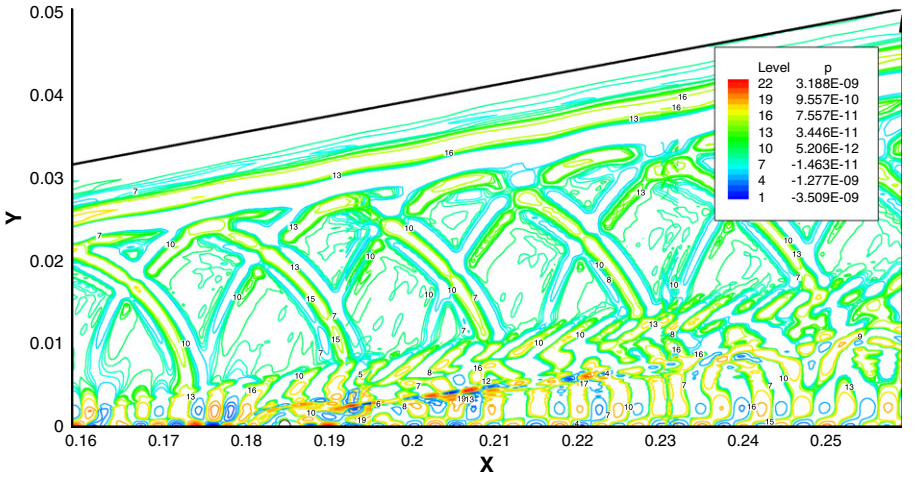

Fig. 17. Contours of pressure perturbations for unsteady Mach 5.92 flow over a flat plate with roughness. The forcing frequency is combined 15 frequencies. The roughness is located at $x_c = 0.185$ m.

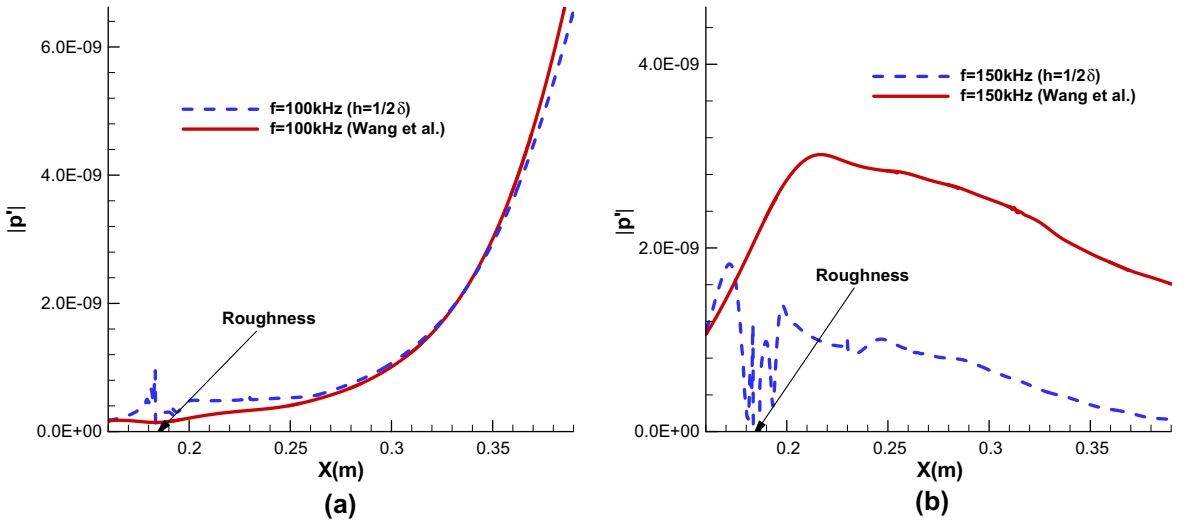

Fig. 18. Comparison of amplitudes of surface pressure perturbations for cases with roughness (dashed line) and without roughness (solid line) at two different frequencies: (a) $f = 100$ kHz, (b) $f = 150$ kHz.

f = 150 kHz, the amplitudes grow in earlier region due to initial transient of the disturbances excited by the wall blowing and suction, followed by a delay. There is no second mode instability in higher frequency. By comparing the wave amplitudes after adding the roughness element, this figure shows that the evolution patterns of disturbance waves are very similar to those of the non-roughness case. But the magnitudes of disturbances vary significantly around the roughness element. For the case of frequency f = 100 kHz, the wave amplitudes increase dramatically as they approach the roughness. The wave magnitudes for the case with roughness remain larger than those of the non-roughness case in further downstream. For the case of f = 150 kHz, however, the wave amplitudes are reduced significantly when the waves pass the roughness as shown in Fig. 18(b). Overall, in the local region of the flow field where the roughness effect to the mean flow is significant, the roughness element also has a strong effect on the unstable disturbances. The disturbance amplitudes may or may not be amplified by the surface roughness, depending on the specific frequency f of the waves. In the further downstream region of the roughness element, the magnitudes of disturbances for all 15 frequencies are lower than those of the non-roughness case. For the case of 100 kHz frequency, the amplitudes of the second mode are stabilized slightly when a roughness element is added into the computational domain. Overall, the current third-order cut-cell method is able to obtain highly accurate results for steady and unsteady viscous hypersonic flow over a flat plate with an isolated surface roughness element.

The current steady flow solution obtained by the current third-order cut-cell method for the hypersonic flow with a roughness element has also been used to conduct a stability analysis of the boundary layer. Since the accuracy of the stability

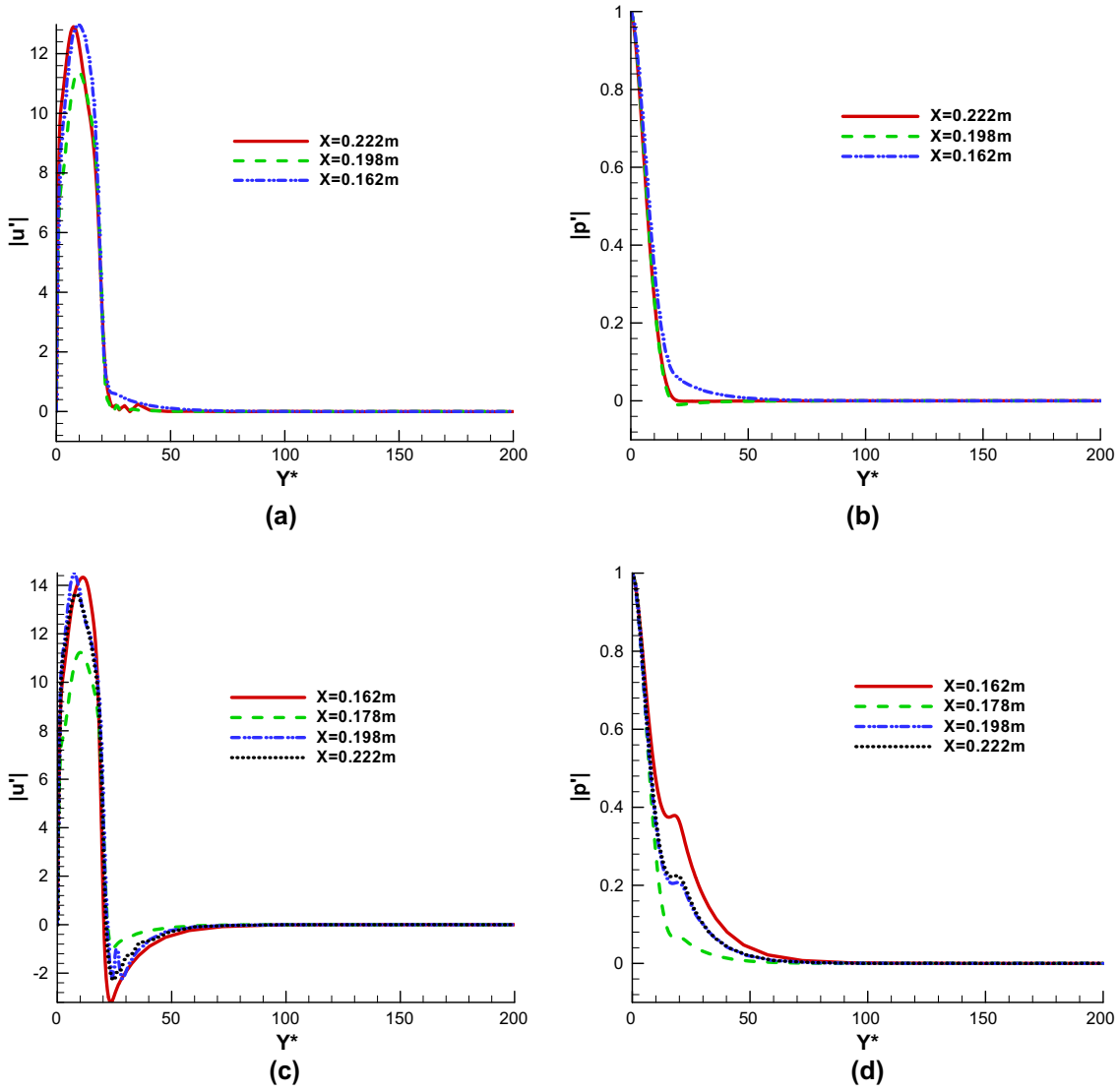

Fig. 19. Wave mode profiles obtained by linear stability analysis at various locations: (a) mode F profile for velocity, (b) mode F profile for pressure, (c) mode S profile for velocity, (d) mode S profile for pressure. The amplitudes of disturbances are nondimensionlized by their corresponding values on the surface of flat plate.

analysis relies on the accuracy of the mean flow solution, the stability results are presented here to demonstrate the accuracy of the current mean flow solution obtained by the cut-cell method.

A multi-domain spectral collocation (MDSC) method of Malik [60] is used to conduct stability analysis of the steady solution for the hypersonic flow over the roughness element. More details about the linear stability analysis and methods can be found in [60].

The dimensionless frequency used for linear stability analysis is defined as

$$F = \frac{2\pi f \nu}{u_\infty^2} \tag{70}$$

where F is dimensionless frequency, ν is kinematic viscous coefficient. In present simulation, $\nu = 6.05 \times 10^{-5}\ \mathrm{m^2/s}$, $F = 5.30 \times 10^{-5}$ for the case of $f = 100$ kHz and $u_\infty = 827.29$ m/s.

In LST analyses of boundary layer flows, the Reynolds number based on the local length scale of boundary layer thickness δ. They are expressed as

$$R = \frac{\rho_\infty u_\infty \delta}{\mu_\infty}, \quad \delta = \sqrt{\frac{\mu_\infty x}{\rho_\infty u_\infty}} \tag{71}$$

Hence the relation between the R and the unit Reynolds number R_∞ is,

$$R = \sqrt{R_\infty x} \tag{72}$$

With the definitions of Reynolds number R and the dimensionless frequency F, the dimensionless circular frequency is expressed as

$$\omega = RF \tag{73}$$

There are two major wave modes in hypersonic boundary layers: mode F and mode S. Fig. 19 shows wave mode profiles obtained by linear stability analysis at various locations for both mode F and mode S. The disturbance amplitudes are non-dimensionlized by the pressure disturbance on the surface of flat plate, e.g. $u'(y) = \tilde{u}(y)/\tilde{p}(0)$. In the upstream of roughness at $x = 0.1620$ m where the roughness effects on the mean flow are not that significant, mode S and mode F can be identified clearly. As we move gradually downstream, the profiles of the wave modes vary significantly. After $x = 0.1980$ m, mode S and mode F can be identified again by examining their perturbation profiles. The mean flow solution obtained by the cut-cell method can produce accuracy results for the stability analysis.

In order to investigate the spatial development of unstable mode quantitatively, a local wave number (α_r) and a local growth rate (α_i) related to pressure perturbation along the flat plate are calculated,

$$\alpha_r = \frac{\delta d\varphi'}{dx} \tag{74}$$

$$\alpha_i = -\frac{\delta d|p'|}{|p'|dx} \tag{75}$$

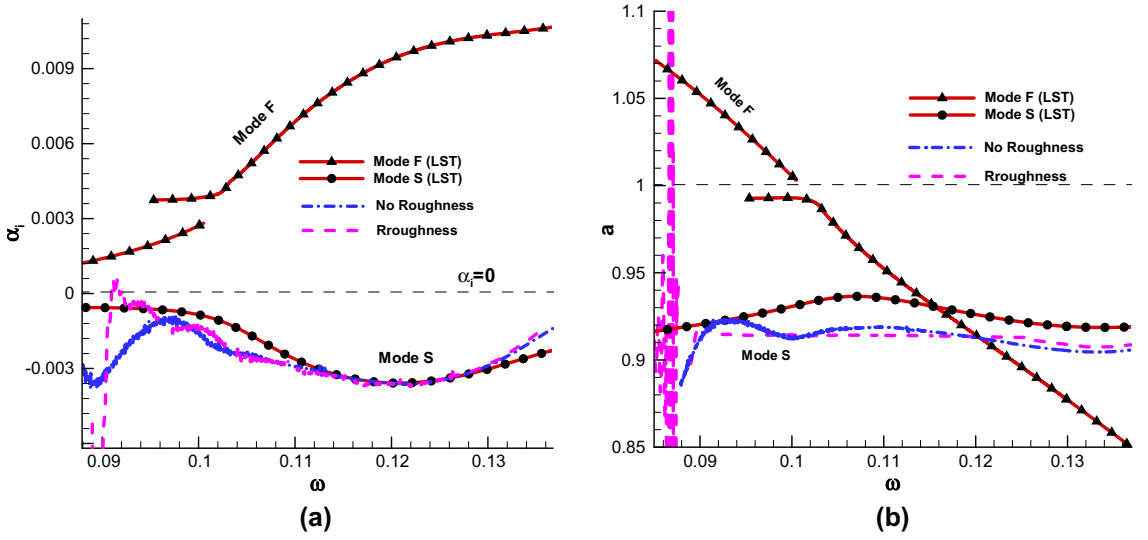

Fig. 20. Comparison of distribution of eigenvalues of mode S and mode F for hypersonic flow over the flat plate with and without the roughness. (a) growth rate α_i, (b) phase velocity $a = \frac{FR}{\alpha_r}$.

where δ is the length scale of local boundary layer thickness as defined by Eq. (71) $|p'|$ and φ' are pressure perturbation amplitude and phase angle, respectively. The parameters α_r and α_i represent the true wave number and growth rate only if the perturbation is dominated by a single wave mode. Otherwise, the disturbance needs to be decomposed in order to check properties of a specific mode. The phase velocity is defined as

$$a = \frac{FR}{\alpha_r} \tag{76}$$

Quantitative comparisons of amplification rates α_i and phase velocities a obtained from DNS and LST are conducted. Fig. 20 show that our numerical results obtained by the cut-cell method agree well with the LST in the region where the parallel flow assumption is valid. Fig. 20(a) compares the growth rate calculated from stability simulations to that from the LST. For mode S, the growth rate of stability simulation with or without surface roughness has a good agreement with that of LST in the region from $\omega = 0.11$ to $\omega = 0.13$. When $\omega > 0.13$, the growth rate of stability simulation is smaller than that of LST, i.e., mode S obtained by stability simulation becomes more stable than that predicted by LST. The discrepancy between the growth rates of mode S is mainly caused by the nonparallel flow effect. Fig. 20(b) compares the phase velocity calculated from stability simulations to that from LST. For mode S, roughness element in the upstream of boundary layer only has minor influence on the phase velocity distribution.

5. Conclusions

In order to overcome the difficulties in generating body-fitted grids hypersonic flow with arbitrary surface roughness, in this paper, we have developed a new high-order cut-cell method to discretize flow equations in an irregular domain. A family of high-order non-uniform-grid finite-difference schemes has been derived to discretize derivatives of the flux terms in the governing equations at irregular points near the cut cell boundary. A dropped-point approach is used to overcome the small cell problem and achieve numerical instability. The new high-order cut-cell method has been tested in the computations of several one and two-dimensional hyperbolic and elliptic equation in irregular domains. The results show that up to fourth-order accuracy in both L^2 and L^∞ norm can be obtained for the current cut-cell method for both problems. We subsequently have applied a third-order cut-cell method to the two-dimensional compressible Navier–Stokes equations for simulating roughness induced receptivity for hypersonic flow over a flat plate with a blowing and suction slot placed near the leading edge. The surface roughness height is approximately half of the local boundary thickness. By using the uniformly third-order cut-cell method, small disturbances generated by the blowing and suction slot are well resolved around the roughness element. The results obtained by the new third-order cut-cell method are consistent with those of the linear stability analysis results. In addition, steady solutions of the flow affected by the roughness are well captured by the cut-cell method. The simulation results suggest that the existence of small surface roughness $(h = \frac{1}{2}\delta)$ affects the propagation of instability waves in the hypersonic boundary layer.

Acknowledgments

This work was sponsored by the Air Force Office of Scientific Research, USAF, under AFOSR Grant # FA9550-07-1-0414, monitored by Dr. John Schmisseur. This work was also sponsored by the AFOSR/NASA National Center, headed by Professor Bill Saric of Taxes A&M University, for Hypersonic Research in Laminar-Turbulent Transition. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements either expressed or implied, of the Air Force Office of Scientific Research or the US Government.

Appendix A

A.1. Second-order non-uniform-grid finite-difference formulas ($p = 2$, $q = 3$)

In this case, the local accuracy for the irregular points is $p = 2$ with a grid stencil of $q = 3$. The local grid stencil contains three points with one irregular point. Hence the set of irregular points is $\Omega_2^\xi = \{\xi_2\}$. The finite-difference stencil for the irregular point ξ_2 consists of the following grids: $\{\xi_1, \xi_2, \xi_3\}$. The corresponding finite-difference coefficients of Eq. (28) are listed in Table A1.

For the second-order simulation, the critical ratio is chosen to be $\Theta = 0.2$, which means a grid point included in the stencil satisfies $\sigma \geqslant 0.2$.

Table A1
Second-order finite-difference coefficients for discretization of viscous terms (where $\sigma = \frac{\eta}{\Delta h}$).

$a_{2,1}$	$a_{2,2}$	$a_{2,3}$
$\frac{-1}{\sigma(1+\sigma)}$	$\frac{1-\sigma}{\sigma}$	$\frac{\sigma}{1+\sigma}$

A.2. Third-order non-uniform-grid finite-difference formulas ($p = 3$, $q = 5$)

In this case, the local accuracy for the irregular points is $p = 3$ with a grid stencil of $q = 5$. The local grid stencil contains five points with two irregular point. Hence the set of irregular points is $\Omega_3^{\xi} = \{\xi_2, \xi_3\}$. The finite-difference stencil for the irregular point ξ_3 consists of the following grids: $\{\xi_1, \xi_2, \xi_3, \xi_4, \xi_5\}$, while that for ξ_2 is $\{\xi_1, \xi_2, \xi_3, \xi_4\}$. The corresponding finite-difference coefficients of Eq. (28) are listed in Table A2.

For the third-order simulation, the critical ratio is chosen to be $\Theta = 0.5$, which means a grid point included in the stencil satisfies $\sigma \geqslant 0.5$. As the order increases, it is necessary to use larger value of Θ in order to maintain numerical stability of the algorithms.

A.3. Fourth-order non-uniform-grid finite-difference formulas ($p = 4$, $q = 6$)

In this case, the local accuracy for the irregular points is $p = 4$ with a grid stencil of $q = 6$. The local grid stencil contains six points with three irregular points. Hence the set of irregular points is $\Omega_4^{\xi} = \{\xi_2, \xi_3, \xi_4\}$. The finite-difference stencil for the irregular point ξ_4 consists of the following grids: $\{\xi_1, \xi_2, \xi_3, \xi_4, \xi_5\}$, while that for both ξ_2 and ξ_3 is $\{\xi_1, \xi_2, \xi_3, \xi_4, \xi_5\}$.

For the fourth-order simulation, the critical ratio is chosen to be $\Theta = 1.0$.

A.4. First-order non-uniform-grid finite-difference formulas for inviscid terms ($p = 1$, $q = 3$)

In this case, the local accuracy for the irregular points is $p = 1$ with a grid stencil of $q = 3$. The local grid stencil contains three points with one irregular point. Hence the set of irregular points is $\Omega_2^{\xi} = \{\xi_2\}$. The finite-difference stencils for the

Table A2
Third-order finite-difference coefficients for discretization of viscous terms.

Coefficients	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$	$a_{i,5}$
$i = 2$	$\frac{-2}{\sigma(1+\sigma)(2+\sigma)}$	$-\frac{3\sigma-2}{2\sigma}$	$\frac{2\sigma}{1+\sigma}$	$\frac{-\sigma}{2(2+\sigma)}$	0
$i = 3$	$\frac{2}{\sigma(1+\sigma)(2+\sigma)(3+\sigma)}$	$-\frac{2(\sigma+1)}{6\sigma}$	$\frac{1-\sigma}{2(1+\sigma)}$	$\frac{1+\sigma}{2+\sigma}$	$\frac{-(1+\sigma)}{6(3+\sigma)}$

Table A3
Fourth-order finite-difference coefficients for discretization of viscous terms.

Coefficients	$a_{i,1}$	$a_{i,2}$	$a_{i,3}$	$a_{i,4}$		$a_{i,5}$	$a_{i,6}$
$i = 2$	$\frac{-6}{\sigma(1+\sigma)(2+\sigma)(3+\sigma)}$	$\frac{-11\sigma+6}{6\sigma}$	$\frac{3\sigma}{1+\sigma}$	$\frac{-3\sigma}{2+\sigma}$		$\frac{\sigma}{3(3+\sigma)}$	0
$i = 3$	$\frac{2}{\sigma(1+\sigma)(2+\sigma)(3+\sigma)}$	$\frac{-2(\sigma+1)}{6\sigma}$	$\frac{-\sigma+1}{2(1+\sigma)}$		$\frac{1+\sigma}{2+\sigma}$	$\frac{-(1+\sigma)}{6(3+\sigma)}$	0
$i = 4$	$\frac{4}{\sigma(1+\sigma)(2+\sigma)(3+\sigma)(4+\sigma)}$	$\frac{-(2+\sigma)}{12\sigma}$	$\frac{2(2+\sigma)}{3(1+\sigma)}$	$\frac{-1}{2+\sigma}$		$\frac{-2(2+\sigma)}{3(3+\sigma)}$	$\frac{2+\sigma}{12(4+\sigma)}$

Table A4
First-order finite-difference coefficients for discretization of inviscid terms.

$b_{2,1}^{+}$	b_2^{+}	b_3^{+}
0	-1	1
b_1^{-}	b_2^{-}	b_3^{-}
$-\frac{1}{\sigma}$	$\frac{1}{\sigma}$	0

Table A5
Second-order finite-difference coefficients for discretization of inviscid terms.

Coefficients	$b_{i,1}^{+}$	$b_{i,2}^{+}$	$b_{i,3}^{+}$	$b_{i,4}^{+}$	$b_{i,5}^{+}$
$i = 2$	0	$-\frac{3}{2}$	2	$-\frac{1}{2}$	0
$i = 3$	0	$-\frac{1}{3}$	$-\frac{1}{2}$	1	$-\frac{1}{6}$
	$b_{i,1}^{-}$	$b_{i,2}^{-}$	$b_{i,3}^{-}$	$b_{i,4}^{-}$	$b_{i,5}^{-}$
$i = 2$	$-\frac{1}{\sigma(1+\sigma)}$	$\frac{1-\sigma}{\sigma}$	$\frac{\sigma}{1+\sigma}$	0	0
$i = 3$	$\frac{1}{\sigma(1+\sigma)(2+\sigma)}$	$-\frac{\sigma+1}{2\sigma}$	$\frac{1}{1+\sigma}$	$\frac{1+\sigma}{2(2+\sigma)}$	0

Table A6
Third-order finite-difference coefficients for discretization of inviscid terms.

Coefficients	$b^+_{i,1}$	$b^+_{i,2}$	$b^+_{i,3}$	$b^+_{i,4}$	$b^+_{i,5}$	$b^+_{i,6}$
$i=2$	0	$-\frac{11}{6}$	3	$-\frac{3}{2}$	$\frac{1}{3}$	0
$i=3$	0	$-\frac{1}{3}$	$-\frac{1}{2}$	1	$-\frac{1}{6}$	0
$i=4$	0	$\frac{1}{12}$	$-\frac{2}{3}$	0	$\frac{2}{3}$	$-\frac{1}{12}$
	$b^-_{i,1}$	$b^-_{i,2}$	$b^-_{i,3}$	$b^-_{i,4}$	$b^-_{i,5}$	$b^-_{i,6}$
$i=2$	$\frac{-2}{\sigma(1+\sigma)(2+\sigma)}$	$\frac{2-3\sigma}{2\sigma}$	$\frac{2\sigma}{1+\sigma}$	$\frac{-\sigma}{2(2+\sigma)}$	0	0
$i=3$	$\frac{1}{\sigma(1+\sigma)(2+\sigma)}$	$-\frac{\sigma+1}{2\sigma}$	$\frac{1}{1+\sigma}$	$\frac{1+\sigma}{2(2+\sigma)}$	0	0
$i=4$	$\frac{-2}{\sigma(1+\sigma)(2+\sigma)(3+\sigma)}$	$\frac{\sigma+2}{6\sigma}$	$-\frac{2+\sigma}{1+\sigma}$	$\frac{4+\sigma}{2(2+\sigma)}$	$\frac{2+\sigma}{3(3+\sigma)}$	0

irregular point ξ_2 consists of the following grids: $\{\xi_2,\xi_3\}$ for F'^+, and $\{\xi_1,\xi_2\}$ for F'^-. The corresponding finite-difference coefficients of Eqs. (32) and (33) are listed in Table A4.

A.5. Second-order non-uniform-grid finite-difference formulas for inviscid terms (p = 2, q = 5)

In this case, the local accuracy for the irregular points is $p=2$ with a grid stencil of $q=5$. The local grid stencil contains five points with two irregular points. Hence the set of irregular points is $\Omega^{\xi}_3=\{\xi_2,\xi_3\}$. The finite-difference stencil for the irregular point ξ_3 consists of the following grids: $\{\xi_2,\xi_3,\xi_4,\xi_5\}$ for F'^+, and $\{\xi_1,\xi_2,\xi_3,\xi_4\}$ for F'^-. The corresponding stencils for ξ_2 are: $\{\xi_2,\xi_3,\xi_4\}$ for F'^+, and $\{\xi_1,\xi_2,\xi_3\}$ for F'^-. The corresponding finite-difference coefficients of Eqs. (32) and (33) are listed in Table A5.

A.6. Third-order non-uniform-grid finite-difference formulas for inviscid terms (p = 3, q = 6)

In this case, the local accuracy for the irregular points is $p=3$ with a grid stencil of $q=6$. The local grid stencil contains six points with three irregular points. Hence the set of irregular points is $\Omega^{\xi}_4=\{\xi_2,\xi_3,\xi_4\}$. The finite-difference stencil for the irregular point ξ_4 consists of the following grids: $\{\xi_2,\xi_3,\xi_4,\xi_5,\xi_6\}$ for F'^+, and $\{\xi_1,\xi_2,\xi_3,\xi_4,\xi_5\}$ for F'^-. The corresponding stencils for ξ_3 are $\{\xi_2,\xi_3,\xi_4,\xi_5\}$ for F'^+, and $\{\xi_1,\xi_2,\xi_3,\xi_4\}$ for F'^-. Those for ξ_2 are: $\{\xi_2,\xi_3,\xi_4,\xi_5\}$ for F'^+, and $\{\xi_1,\xi_2,\xi_3,\xi_4\}$ for F'^-. The corresponding finite-difference coefficients of Eqs. (32) and (33) are listed in Table A6.

References

[1] S.P. Schneider, Summary of hypersonic boundary-layer transition experiments on blunt bodies with roughness, Journal of Spacecraft and Rockets 45 (2008) 1090–1112.
[2] D.S. Board, Final report of the second defense science board task force on the national aero-space plane (NASP), 1992, p. 94-00052.
[3] J.D. Anderson, Hypersonic and high temperature gas dynamics, in: AIAA, 2000.
[4] S. Berry, T. Horvath, Discrete roughness transition for hypersonic flight vehicles, in: 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2007.
[5] W.S. Saric, H.L. Reed, E.J. Kerschen, Boundary-layer receptivity to freestream disturbances, Annual Review of Fluid Mechanics 34 (2002) 291–319.
[6] A.H. Boudreau, Artificial induced boundary-layer transition on blunt-slender cones at hypersonic speeds, Journal of Spacecraft and Rockets 16 (4) (1979) 245–251.
[7] K. Mahesh, A family of high order finite difference schemes with good spectral resolution, Journal of Computational Physics 145 (1998) 332–358.
[8] T.J. Poinsot, S.K. Lele, Boundary conditions for direct simulations of compressible viscous flow, Journal of Computational Physics 101 (1) (1992) 104–129.
[9] C.D. Pruett, C-L. Chang, Spatial direct numerical simulation of high-speed boundary-layer flows Part II: Transition on a cone in Mach 8 flow, Theoretical and Computational Fluid Dynamics 7 (1995) 397–424.
[10] S.K. Lele, Compact finite difference schemes with spectral-like resolution, Journal of Computational Physics 103 (1992) 16–42.
[11] N.A. Adams, K. Shariff, A high-resolution hybrid compact-ENO scheme for shock-turbulence interaction problems, Journal of Computational Physics 127 (1996) 27–51.
[12] D.V. Gaitonde, M.R. Visbal, Further development of a Navier–Stokes solution procedure based on higher-order formulas, AIAA Paper 1999-0557, 1999.
[13] X. Zhong, High-order finite-difference schemes for numerical simulation of hypersonic boundary-layer transition, Journal of Computational Physics 144 (1998) 662–709.
[14] X. Zhong, Leading-edge receptivity to free stream disturbance waves for hypersonic flow over a parabola, Journal of Fluid Mechanics 441 (2001) 315–367.
[15] R.P. Fedkiw, T.D. Aslam, S. Xu, The ghost fluid method for deflagration and detonation discontinuities, Journal of Computational Physics 154 (2) (1999) 393–427.
[16] H.S. Udaykumar, R. Mittal, W. Shyy, Computation of solid–liquid phase fronts in the sharp interface limit on fixed grids, Journal of Computational Physics 153 (1999) 534–574.
[17] H.S. Udaykumar, R. Mittal, P. Rampunggoon, A. Khanna, A sharp interface cartesian grid method for simulating flows with complex moving boundaries, Journal of Computational Physics 174 (2001) 345–380.
[18] C. Peskin, The immersed boundary method, Acta Numerica (2002) 479–527.
[19] L. Lee, R. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM Journal of Scientific Computing 25 (2003) 832–856.
[20] O. Marxen, G. Iaccarino, Numerical simulation of the effect of a roughness element on high-speed boundary-layer instability, in: 38th Fluid Dynamics Conference and Exhibit, Seattle, Washington, 2008.

[21] H. Johansen, P. Colella, A cartesian grid embedded boundary method for Poisson's equation on irregular domains, Journal of Computational Physics 147 (1998) 60–85.
[22] P. McCorquodale, P. Colella, H. Johansen, A cartesian grid embedded boundary method for the heat equation on irregular domains, Journal of Computational Physics 173 (2001) 620–635.
[23] F. Gibou, R.P. Fedkiw, A fourth order accurate discretization for the Laplace and heat equations on arbitrary domain with applications to the Stefan problem, Journal of Computational Physics 202 (2005) 577–601.
[24] D.M. Causon, D.M. Ingram, C.G. Mingham, A Cartesian cut cell method for shallow water flows with moving boundaries, Advances in Water Resources 24 (2001) 899–911.
[25] M.A. Dumett, J.P. Keener, An immersed interface method for solving anisotropic elliptic boundary value problems in three dimensions, SIAM Journal of Scientific Computing 25 (2003) 348–367.
[26] Z. Li, Immersed interface methods for moving interface problem, Numerical Algorithms 14 (1997) 269–293.
[27] Z. Li, A fast iterative algorithm for elliptic interface problems, SIAM Journal on Numerical Analysis 35 (1998) 230–254.
[28] A. Wiegmann, K.P. Bube, The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions, SIAM Journal on Numerical Analysis 37 (1998) 827–862.
[29] A. Wiegmann, K.P. Bube, The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions, SIAM Journal on Numerical Analysis 37 (2000) 827–862.
[30] H. Huang, Z. Li, Convergence analysis of the immersed interface method, IMA Journal of Numerical Analysis 19 (1999) 583–608.
[31] A.L. Fogelson, J.P. Keener, Immersed interface methods for Neumann and related problems in two and three dimensions, SIAM Journal of Scientific Computing 22 (2000) 1630–1654.
[32] K. Ito, Z. Li, Solving a nonlinear problem in magneto-rheological fluids using the immersed interface method, Journal of Computational Physics 19 (2003) 253–266.
[33] S. Deng, K. Ito, Z. Li, Three-dimensional elliptic solvers for interface problems and applications, Journal of Computational Physics 184 (2003) 215–243.
[34] L. Adams, T.P. Chartier, New geometric immersed interface multigrid solvers, SIAM Journal of Scientific Computing 25 (2004) 1516–1533.
[35] X. Zhong, A new high-order immersed interface method for multi-phase flow, AIAA paper 2006-1294, 2006.
[36] R. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM Journal of Numerical Analysis 31 (1994) 1001–1025.
[37] R.J. LeVeque, Z. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM Journal of Scientific Computing 18 (1997) 709–735.
[38] M.J. Berger, R.J. Leveque, Stable boundary condition for cartesian grid calculations, Computer System in Engineering 1 (1990) 305–311.
[39] J.J. Quirk, An alternative to unstructured grids for computing gas dynamic flows around arbitrarily complex bodies, Computers and Fluids 23 (1994) 125–142.
[40] P. Balakumar, Transition in a supersonic boundary-layer due to roughness and acoustic disturbances, in: AIAA, 2003.
[41] M. Choudhari, P. Fischer, Roughness-induced transient growth, in: 35th AIAA Fluid Dynamics Conference, 2005.
[42] S. Ghosh, M. Hossain, W.H. Matthaeus, The application of spectral methods in simulating compressible fluid and magnetofluid turbulence, Computer Physics Communications 74 (1993) 18–40.
[43] A. Zang, Spectral methodsnext term for simulations of previous termtransitionnext term and turbulence, Computer Methods in Applied Mechanics and Engineering 80 (1–3) (1990) 209–221.
[44] H.G. Weller, C.J. Marooney, A.D. Gosman, A new spectral method for calculation of the time-varying area of a laminar flame in homogeneous turbulence, in: Symposium on Combustion, vol. 23, 1991, pp. 629–636.
[45] M.a.M.P. Rai, Direct numerical simulation of transition and turbulence in a spatially evolving boundary layer, Journal of Computational Physics 109 (1993) 169–192.
[46] H. Fasel, A. Thumm, H. Bestek, Direct numerical simulation of transition in supersonic boundary layers: oblique breakdown, in: Fluids Engineering Conference on Transitional and Turbulent Compressible Flows, Washington, DC, June 20–24, 1993, ASME, New York, 1993.
[47] N.A. Adams, L. Kleiser, Subharmonic transition to turbulence in a flat-plate boundary layer at Mach number 4.5, Journal of Fluid Mechanics 317 (1996) 301–335.
[48] C.D. Pruett, Spatial Direct Numerical Simulation of Transitioning High-Speed Flows, in: L.D. Kral, E.F. Spina, C. Arakawa (Eds.), Transitional and Turbulent Compressible Flows, FED-vol. 224, ASME, 1995, pp. 63–70.
[49] G.S. Jiang, C.W. Shu, Efficient implementation of weighted ENO schemes, Journal of Computational Physics 126 (1) (1996) 202–228.
[50] M.Y. Hussaini, D.A. Kopriva, M.D. Salas, T.A. Zang, Spectral methods for the Euler equations. Part II. Chebyshev methods and shock fitting, AIAA Journal 23 (1985) 234–240.
[51] A.A. Maslov, A.N. Shiplyuk, A. Sidorenko, D. Arnal, Leading-edge receptivity of a hypersonic boundary layer on a flat plate, Journal of Fluid Mechanics 426 (2001) 73–94.
[52] C.H. Lee, Y.H. Chu, A new type of TVD schemes for computations of high speed flows, in: AIAA, 1993.
[53] Y. Ma, X. Zhong, Receptivity of a supersonic boundary layer over a flat plate. Part 1: Wave structures and interactions, Journal of Fluid Mechanics 488 (2003) 31–78.
[54] X. Wang, X. Zhong, Numerical simulation and experiment comparison of leading-edge receptivity of a Mach 5.92 boundary layer, in: AIAA paper 2006-1107, 2006.
[55] A. Tumin, X. Wang, X. Zhong, Direct numerical simulation and the theory of receptivity in a hypersonic boundary layer, Physics of Fluids 19 (2007) 1–14.
[56] L.M. Mack, Boundary layer linear stability theory, in: AGARD Report, 1984, pp. 1–81.
[57] A.A. Maslov, S.G. Mironov, A.A. Shiplyuuk, A.A. Sidorenko, D.A. Buntin, V.M. Aniskin, Hypersonic flow stability experiments, in: AIAA 2002-0153, 2002.
[58] A. Demetriades, Hypersonic viscous flow over a slander cone. Part III: Laminar instability and transition, in: AIAA paper 74-535, 1974.
[59] A. Demetriades, Laminar boundary layer stability measurements at Mach 7 including wall temperature effects, AFOSR-TR-77-1311, 1977.
[60] M.R. Malik, Numerical methods for hypersonic boundary layer stability, Journal of Computational Physics 86 (1990) 376–413.