

The variable-extended immersed boundary method for compressible gaseous reactive flows past solid bodies

Jian-Hang Wang¹ | Chi Zhang

Chair of Aerodynamics and Fluid Mechanics, Department of Mechanical Engineering, Technical University of Munich, Garching, Germany

Correspondence

Jian-Hang Wang and Chi Zhang, Chair of Aerodynamics and Fluid Mechanics, Department of Mechanical Engineering, Technical University of Munich, Garching 85748, Germany.
Email: jianhang.wang@tum.de and c.zhang@tum.de

Funding information

EU Marie Skłodowska-Curie Innovative Training Networks, Grant/Award Number: 675528-IPPAD-H2020-MSCA-ITN-2015

Abstract

An immersed boundary method (IBM) has been developed to handle the solid body embedded flowfield simulation for compressible reactive flows, paving the way of application for a wide range of fluid–solid interaction problems. Previously, the Brinkman penalization method (BPM), originated from porous media flows, has been successfully used for incompressible Navier–Stokes equations by adding penalization terms to momentum equations. However, it is nontrivial to solve the compressible form due to the penalized continuity equation that usually poses severe numerical stiffness. In order to circumvent this issue, an extending procedure for relevant variables from the fluid to solid domain is considered, by analyzing the ordinary differential equations remained after operator splitting. Density can be then determined with the help of an equation of state (EoS). Meanwhile, efforts of enforcing the Neumann boundary condition, for example, the adiabatic wall condition, on the fluid–solid interface can be minimized by extending temperature across the interface directly. One more advantage of the extending step lies in that it can quickly reach a steady state when performed within a narrow band around the interface. Implemented into an adaptive Cartesian grid-based flow solver for compressible Navier–Stokes equations with chemical reaction source terms, the present variable-extended IBM is validated by numerical examples ranging from single-species nonreactive to multispecies detonative flows in one- and two-dimensional domains. Numerical results show (1) the successful specification of slip or nonslip, adiabatic or isothermal wall condition on the fluid–solid interface and (2) loss of total energy in the original BPM being avoided and the numerical accuracy being improved especially for energy-sensitive reactive flows.

KEYWORDS

adaptive refinement, Brinkman penalization, detonation, extending equation, immersed boundary method, reactive flow

1 | INTRODUCTION

Simulation of internal flows in many air-breathing propulsion systems or external flows past reentry vehicles, and so forth, must deal with solid bodies embedded in the fluid domain, which is heavily challenged by the difficulties of treating

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2021 The Authors. *International Journal for Numerical Methods in Engineering* published by John Wiley & Sons Ltd.

complex geometries and the associated boundary conditions. Two main categories of methods are usually considered to handle complex geometries: body-fitted grid-based methods and immersed boundary methods (IBMs). Body-fitted methods rely on structured or unstructured grids that are conformal to the complex boundaries. To generate a mesh of good quality, one should be very careful even for some simple geometries, which leads to the mesh generation process being prohibitively expensive. Besides, for a given spatial discretization scheme, the order of accuracy on structured or unstructured grids is always lower than on Cartesian grids.¹ Alternatively, one can choose to perform simulations on simple Cartesian grids and to impose immersed boundary conditions upon the fluid. Two essential advantages of this idea are its easy implementation and relatively straightforward extension from stationary body to more general moving body cases without mesh regeneration.

A large group of IBMs belong to the direct-forcing IBM. It on one hand mimics the behavior of body-fitted methods that use ghost cells, and imposes the boundary conditions directly on the immersed boundaries.^{2–8} Multidimensional interpolation or extrapolation in the vicinity of the fluid–solid interface is usually needed in order to determine the image-point state in various cases and thus the ghost-cell state, making use of intersection points on the interface to impose exact boundary conditions.⁹ On the other hand, some cut-cell methods^{10–13} confront the obstacles about the specification of the type or local geometry of a cut cell (especially in high-dimensional situations) as well as the fact that small cut cells might lead to accuracy losses and/or strict timestep restrictions.¹⁴

As another type of IBMs, the Brinkman penalization method (BPM) solves the fluid and the solid domain simultaneously by introducing fictitious (penalization) terms in a unified system of governing equations. In this way, there is no need to impose the interface conditions explicitly since they are automatically solved from the governing equations. The BPM originated from the idea of considering the solid body as a porous medium with a very low permeability and the fluid as a medium with an infinite permeability by Arquis and Caltagirone,¹⁵ and became available for incompressible Navier–Stokes equations first by Angot et al.¹⁶ Liu and Vasilyev¹⁷ started to extend the BPM to the regime of compressible viscous flows by penalizing the continuity equation in addition to the momentum and energy equations. They also pointed out that only penalizing the momentum and energy equations can produce unsatisfactory results, mostly due to nonphysical wave transmissions into solid bodies, resulting in considerable energy and mass losses in reflected waves. Boiron et al.¹⁴ further considered the BPM for large Mach number flows. Anand et al.¹⁸ investigated the BPM in combination with a high-order discontinuous Galerkin scheme, and stated that the modeling error (e.g., $\mathcal{O}(\eta^{1/2}\phi)$ for resolved boundary layers¹⁷) can be minimized with sufficiently small permeability η when holding the porosity $\phi = 1$. Piquet et al.¹ utilized an operator-splitting scheme to separately treat the penalization terms in a semiimplicit manner. All the works above focused the discussion on the isothermal wall condition or Dirichlet boundary condition for temperature or total energy. Besides, to apply the Neumann boundary condition for temperature, Brownkymkoski et al.¹⁹ proposed a characteristic-based volume penalization (CBVP) method that takes into account adiabatic walls and mixed boundary conditions. With CBVP, Hardy et al.²⁰ simulated weakly compressible reacting gas-particle flows using a predictor–corrector scheme on a staggered mesh. In a general sense, severe numerical stiffness in the penalized continuity equation and nontrivial imposition of the Neumann boundary condition are two main bottlenecks in BPM.^{9,19}

Aiming at the simulation of fully compressible gaseous reactive flows past solid bodies, the present study begins with the penalized compressible Navier–Stokes equations with chemical reaction source terms, including the penalization of the continuity equation. To overcome the numerical stiffness resulted from the penalized continuity equation, an extending step for relevant primitive variables from the fluid to solid domain is considered to replace solving the ordinary differential equations (ODEs) of the penalization terms after operator splitting. Density inside the solid body can be then determined with the help of an EoS. Meanwhile, by extending temperature across the interface directly, which is analogous to the treatment in the body-fitted methods, the Neumann boundary condition (e.g., the adiabatic wall condition) can be applied in a considerably simple and robust manner. Special attention to imposing the velocity condition on the immersed boundary is also paid to avoid the loss of total energy by the original BPM. To minimize the computational cost and numerical complexity, the present extending operation on one hand is only performed within a narrow band around the fluid–solid interface in the normal direction from the fluid to solid domain; it on the other hand can converge to a smooth steady-state solution unconditionally, given the nature of the advection equation (i.e., the extending equation) with unit velocity along the normal direction. Having implemented the present method into an adaptive Cartesian grid-based flow solver for compressible reactive Navier–Stokes equations, we carry out a variety of numerical examples from single-species nonreactive to multispecies detonative flows in one- and two-dimensional (1D and 2D) cases.

The article is organized as follows. In Section 2, we first introduce the fully compressible Navier–Stokes equations with chemical reaction source terms as well as boundary conditions on the fluid–solid interface. Then the BPM is considered to

enforce such boundary conditions into the compressible reactive Navier–Stokes equations. The present variable-extended IBM is followed. In addition, we specify the spatial and temporal discretization schemes in our adaptive Cartesian grid-based flow solver, and only consider one-way fluid–solid interaction in the scope of this study since the extension to moving body dynamics is straightforward. In Section 3, we examine the present method by extensive numerical examples. Conclusions are drawn in the last section.

2 | METHODOLOGY

2.1 | Governing equations

Let Ω be the computational domain containing N fixed regular obstacles ω_s^n , $1 \leq n \leq N$, and we set

$$\Omega_s = \bigcup_{n=1}^N \omega_s^n \quad \text{and} \quad \Omega_f = \Omega \setminus \Omega_s. \tag{1}$$

Here, Ω_s is the closed region occupied by the solid bodies and Ω_f is the fluid domain.

For the fluid domain Ω_f , we consider the fully compressible Navier–Stokes equations coupled with gaseous finite-rate nonequilibrium chemistry, together with appropriate boundary conditions on the solid body surface (or fluid–solid interface) $\partial\omega_s^n$ and on the boundary of the computational domain $\partial\Omega$. For simplicity and without loss of generality, the system of equations in 2D space is considered and has the conservative form

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = \frac{\partial F_v}{\partial x} + \frac{\partial G_v}{\partial y} + S, \tag{2}$$

where

$$\begin{aligned}
 U &= \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_t \\ \rho y_1 \\ \rho y_2 \\ \vdots \\ \rho y_{ns-1} \end{pmatrix}, F = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ (\rho e_t + p)u \\ \rho y_1 u \\ \rho y_2 u \\ \vdots \\ \rho y_{ns-1} u \end{pmatrix}, G = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ (\rho e_t + p)v \\ \rho y_1 v \\ \rho y_2 v \\ \vdots \\ \rho y_{ns-1} v \end{pmatrix}, \text{ and} \\
 F_v &= \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} - q_x \\ -\rho y_1 \hat{u}_1 \\ -\rho y_2 \hat{u}_2 \\ \vdots \\ -\rho y_{ns-1} \hat{u}_{ns-1} \end{pmatrix}, G_v = \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{xy} + v\tau_{yy} - q_y \\ -\rho y_1 \hat{v}_1 \\ -\rho y_2 \hat{v}_2 \\ \vdots \\ -\rho y_{ns-1} \hat{v}_{ns-1} \end{pmatrix}, S = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \dot{\omega}_1 \\ \dot{\omega}_2 \\ \vdots \\ \dot{\omega}_{ns-1} \end{pmatrix} \tag{3}
 \end{aligned}$$

are the vector of conserved variables, the inviscid flux vectors, the viscous flux vectors in the x and y directions and chemical reaction source terms, respectively. Here ρ , $\mathbf{u} = [u, v]^T$, p , T , and e_t denote the density, velocity, pressure, temperature, and the specific total energy, respectively, and $e_t = e + \frac{1}{2}(u^2 + v^2)$ including the specific internal energy e . y_i is the mass fraction of the i th species in the ns -species reactive gas mixture, and constrained by mass positivity and conservation, that

is, $y_i \geq 0$ and $\sum_{i=1}^{ns} y_i = 1$. In the viscous flux vectors, τ_{ij} is the shear stress defined by

$$\begin{aligned}\tau_{xx} &= \frac{2}{3}\mu \left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} \right), \\ \tau_{xy} &= \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right), \\ \tau_{yy} &= \frac{2}{3}\mu \left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} \right),\end{aligned}\quad (4)$$

with μ being the dynamic viscosity of the fluid. Viscosity can be calculated using Sutherland's law for single species in combination with a proper mixing rule such as the Wilke's semiempirical formula.²¹ The heat flux is

$$\begin{aligned}q_x &= -k\frac{\partial T}{\partial x} + \rho \sum_{i=1}^{ns} h_i y_i \hat{u}_i, \\ q_y &= -k\frac{\partial T}{\partial y} + \rho \sum_{i=1}^{ns} h_i y_i \hat{v}_i,\end{aligned}\quad (5)$$

where k is the thermal conductivity and molecular diffusion terms can be approximated by using

$$\begin{aligned}y_i \hat{u}_i &= -\frac{\mu}{\rho Sc} \frac{\partial y_i}{\partial x}, \\ y_i \hat{v}_i &= -\frac{\mu}{\rho Sc} \frac{\partial y_i}{\partial y},\end{aligned}\quad (6)$$

with $Sc = 0.5$ being the Schmidt number. In the source terms, $\dot{\omega}_i$ represents rate of change of the i th species concentration due to nr finite-rate chemical reactions, that is,

$$\sum_{i=1}^{ns} v_{ji}^f X_i \Leftrightarrow \sum_{i=1}^{ns} v_{ji}^b X_i, \quad j = 1, 2, \dots, nr, \quad (7)$$

and we have

$$\dot{\omega}_i = W_i \sum_{j=1}^{nr} (v_{ji}^b - v_{ji}^f) \left[k_j^f \prod_{l=1}^{ns} \left[\frac{\rho_l}{W_l} \right]^{v_{jl}^f} - k_j^b \prod_{l=1}^{ns} \left[\frac{\rho_l}{W_l} \right]^{v_{jl}^b} \right], \quad (8)$$

with $\rho_l = \rho y_l$. When the flow is inert or nonreactive without activating chemical reactions, source terms are replaced by a zero vector.

To close the system, an EoS of the form

$$p = \rho \sum_{i=1}^{ns} y_i \frac{R_u}{W_i} T \quad (9)$$

is used for the ideal gas mixture, with W_i denoting the molecular weight of the i th species and R_u the universal gas constant.

On the surface of each solid body ω_s^n , the fluid velocity should satisfy the no-slip condition

$$\mathbf{u}|_{\partial\omega_s^n} = \mathbf{u}_s^n, \quad (10)$$

given the moving solid body velocity \mathbf{u}_s^n . $\mathbf{u}_s^n = \mathbf{0}$ corresponds to a stationary solid body. For variables such as density, pressure, temperature, species mass fractions, and so forth, the Dirichlet boundary condition as

$$\varphi|_{\partial\omega_s^n} = \varphi_s^n, \quad (11)$$

(given the value of any variable φ on the solid body surface), or the Neumann boundary condition as

$$\left. \frac{\partial \varphi}{\partial \mathbf{n}} \right|_{\partial \omega_s^n} = \alpha, \quad (12)$$

is usually imposed. For example, isothermal and adiabatic wall boundary condition for temperature on the solid body surface correspond to these two cases, respectively, with $\alpha = 0$ in the latter one.

2.2 | Brinkman penalization method

One very easy-to-implement method to enforce the nonslip velocity condition on solid body surfaces in IBMs is BPM, which considers the solid body as a porous medium with a very low permeability and the fluid as a medium with an infinite permeability. It was originally developed for simulating incompressible flows¹⁶ and imposing the Dirichlet boundary condition for other variables in addition to the nonslip condition for velocity in Equation (10). For the fully compressible Navier–Stokes equations, Brinkman penalization terms are added, giving

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + P = \frac{\partial F_v}{\partial x} + \frac{\partial G_v}{\partial y} + S, \quad (13)$$

and

$$P = \frac{\chi}{\eta} \begin{pmatrix} \left(\left(\frac{1}{\phi} - 1 \right) \left(\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} \right) \right) \\ \rho u - (\rho u)_s \\ \rho v - (\rho v)_s \\ \rho e_t - (\rho e_t)_s \\ \rho y_1 - (\rho y_1)_s \\ \rho y_2 - (\rho y_2)_s \\ \vdots \\ \rho y_{ns-1} - (\rho y_{ns-1})_s \end{pmatrix}, \quad (14)$$

where ϕ is the porosity with $0 < \phi \ll 1$, η is the permeability with $0 < \eta \ll 1$ and

$$\chi = \begin{cases} 1, & \text{if } \mathbf{x} \in \Omega_s, \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

is the characteristic (or mask) function of the solid domain Ω_s . It is readily to see that the penalization terms are effective only in the solid domain with $\chi = 1$. It is worth noticing that the above system of equations is solved for both the fluid and the solid domain. Using the mask function χ , the two domains are automatically defined and treated in a smooth and unified manner. This is a merit of BPM since it inherently skipped the geometric details of the fluid–solid interface inside a cut cell and avoids dealing with the various types of cut cells in high-dimensional situations with other IBMs. Extensive studies^{17,22} have shown the controllable convergence of the penalization method by prior parameters such as by decreasing ϕ and η , or increasing the interface grid resolution.

Based on operator splitting, the penalization terms (\mathbf{P}) can be separated from the convection-diffusion reaction terms (\mathbf{C}).¹ For example, considering the second-order Strang splitting,²³ we have the following fractional steps as

$$U^{n+1} = \mathbf{P} \left(\frac{\Delta t}{2} \right) \mathbf{C}(\Delta t) \mathbf{P} \left(\frac{\Delta t}{2} \right) U^n. \quad (16)$$

Therefore, in terms of the penalization step, a series of ODEs are left, for example,

$$\frac{d}{dt} \begin{pmatrix} \rho u \\ \rho v \\ \rho e_t \end{pmatrix} = -\frac{1}{\eta} \begin{pmatrix} \rho u - (\rho u)_s \\ \rho v - (\rho v)_s \\ \rho e_t - (\rho e_t)_s \end{pmatrix} \quad (17)$$

for the momentum and energy equations, and an analytical solution exists for such a type of ODE¹ as

$$q^{n+1} = q_s \left(1 - e^{-\frac{\Delta t}{\eta}} \right) + q^n e^{-\frac{\Delta t}{\eta}} \quad (18)$$

for a time interval $\Delta t = t^{n+1} - t^n$, in spite of that η might be a very small number. The penalized continuity equation, however, takes a more complicated form if $\phi \neq 1$ according to Equation (14) as

$$\frac{\partial \rho}{\partial t} + \frac{1}{\eta} \left(\frac{1}{\phi} - 1 \right) \left(\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} \right) = 0. \quad (19)$$

Since $0 < \phi \ll 1$ leads to $\frac{1}{\phi} \gg 1$, the penalized continuity equation might exhibit very severe numerical stiffness in terms of a finite timestep of Δt . In such a case, a very tiny local timestep on the solid body surface is necessary to overcome the stiffness, which inevitably limits the size of the global timestep and leads to the low overall computational efficiency. Therefore, some studies^{14,18} simply eliminate the penalization terms in the continuity equation by assuming $\phi = 1$ at the cost of losing some numerical accuracy. It is also possible to use implicit schemes to deal with the numerical stiffness while the increased computational complexity must be compromised with the decreasing size of ϕ . In some practice, the CFL number cannot yet be greater than 1.0 even though the implicit scheme such as LU-SGS²⁴ is employed for reactive flows. The severe numerical stiffness in the penalized continuity equation is one disadvantage of BPM.⁹ Another one lies in the nontrivial application of the Neumann boundary condition in Equation (12).¹⁹

2.3 | Variable-extended IBM

With the help of Equation (9), we see that the penalized density can be obtained directly through the EoS using other primitive variables such as pressure, temperature and mass fractions, instead of solving the stiff penalized continuity equation in Equation (19). In Reference 19, the penalization equation for temperature with the Neumann boundary condition (e.g., $\alpha = 0$ in Equation (12)) in the solid domain ($\chi = 1$) is given as

$$\frac{\partial T}{\partial t} + \frac{1}{\eta} \left(\frac{\partial T}{\partial \mathbf{n}} \right) = 0, \quad (20)$$

where $\mathbf{n} = [n_x, n_y]^T$ is the unit normal vector of the solid body surface pointing outward to the fluid domain. We notice that Equation (20) is similar with the extending equation in ghost fluid methods^{25,26} that has the form of an advection equation

$$\frac{\partial T}{\partial \tau} + \frac{\partial T}{\partial \mathbf{n}} = 0, \quad (21)$$

with τ being a pseudo time here. It is readily to see that a steady state, corresponding to $\frac{\partial T}{\partial \mathbf{n}} = 0$ on the solid body surface and inside the solid domain, can be derived from the extending equation and is one solution to Equation (20), regardless of how small the size of η is or how relatively large the size of Δt is. More extending equations are available for pressure and mass fractions if the Neumann boundary condition is considered. Note that the direction of extending should be pointed from the fluid domain towards the solid domain.

Remark 1. The extending procedure resembles the ghost-cell methods that need to populate the ghost-cell states by reflecting the interpolated image-point states inside the fluid domain in combination with the exact boundary condition of the

intersection points on the fluid–solid interface. The numerical complexity results from the interpolation (or extrapolation) schemes and the treatment of different types of cut cells along with intersection points. However, the present extending operation derives from the penalization ODE after operator splitting, and it can converge to a smooth steady-state solution unconditionally. It therefore spares the efforts of considering the geometric complexity of local interface especially in the existence of cut cells, image points and intersection points, and inherits such a critical merit of BPM.

Solving the extending equation such as Equation (21) is computationally inexpensive with respect to the narrow band of ghost cells on the solid body surface and inside the solid domain, as depicted in Figure 1. The width of the narrow band mainly depends on the length of the interpolation stencil for high-order reconstruction schemes in the calculation of inviscid fluxes. For example, given the ghost cell (i, j) which is in direct contact with two fluid cells and the local normal vector \mathbf{n} in the figure, one can easily find the extending relies on the neighboring cells $(i - 1, j)$ and $(i, j + 1)$. In general, considering a linear approximation, we have the explicit expression

$$T_{ij}^{n+1} = \frac{f_i \left(T_{i-1,j}^n, T_{i+1,j}^n, n_x \right) |n_x| + f_j \left(T_{i,j-1}^n, T_{i,j+1}^n, n_y \right) |n_y|}{|n_x| + |n_y|}, \quad (22)$$

where

$$f_i \left(T_{i-1,j}^n, T_{i+1,j}^n, n_x \right) = \begin{cases} T_{i-1,j}^n, & \text{if } n_x < 0, \\ T_{i+1,j}^n, & \text{otherwise,} \end{cases}$$

$$f_j \left(T_{i,j-1}^n, T_{i,j+1}^n, n_y \right) = \begin{cases} T_{i,j-1}^n, & \text{if } n_y < 0, \\ T_{i,j+1}^n, & \text{otherwise.} \end{cases} \quad (23)$$

For the remaining cells not in direct contact with fluid cells, using Equation (22) in an iterative manner will result in the desired steady state quickly within the narrow band. The computational cost of such an extending procedure over the narrow-band area only is much less than solving the penalized Navier–Stokes equations over the entire solid domain in the original penalization method.

Remark 2. The iterative algorithm here is robust and can maintain the smoothness of the flowfield extended from the fluid domain to the solid domain around the interface. The resulting accuracy is easily adjustable by using different schemes to approximate the gradient term in Equation (21).

Similar with extending temperature from the fluid domain to the solid domain, pressure and mass fractions of species can be also extended subsequently. The EoS will then supplement other variables needed in the ghost cells and a complete set of data used for the inviscid/viscous flux calculation can be constituted.

One more concern in the compressible reactive flow simulation exists, that is, even though the penalized velocity inside the solid body will quickly recover the solid body velocity as in Equation (10), it in fact yields a certain loss of

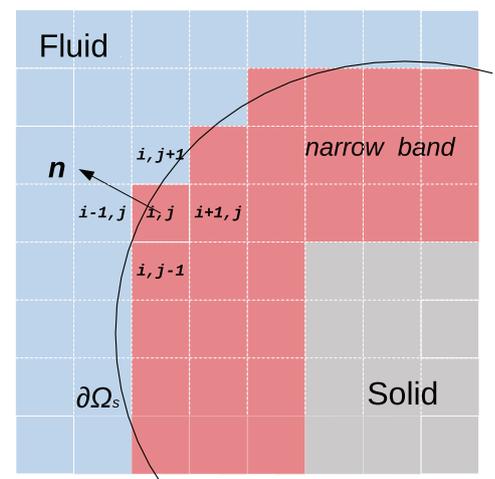


FIGURE 1 Schematic of the fluid and the solid domain including the narrow band of cells on the solid body surface and inside the solid domain, which are also the ghost cells for the fluid domain

the total energy. The loss of energy does not matter much for incompressible or weakly compressible flows. However, for compressible flows, especially for the energy-sensitive reactive flows, such a loss possibly gives rise to inaccurate time-dependent solutions, for example, a slowly propagating reacting front which will be given a description in a following numerical example. Using Figure 2(A) and assuming the solid wall is stationary ($u_w = 0$), we can find that while the interior cell in the fluid domain (i, j) has a positive velocity of $u_i > 0$, its image ghost cell ($i + 1, j$) inside the solid wall ought to have a velocity of $u_g = 0$ by the penalization method. However, we can recall that in the body-fitted grid-based method, an exact nonslip wall condition essentially gives $u_g = -u_i$ such that the total energy $e_t = e + \frac{1}{2}u^2$ on both sides of the wall equals each other. In the penalization method, an energy loss of $\frac{1}{2}u^2$ is a by-product, which is likely to contaminate the accuracy of the compressible reactive flow simulation. Therefore, we continue to replace the penalization step by an extending step for velocity, similar with that for temperature in Equation (22), to have $\mathbf{u}_{i+1,j}^{n+1}$ in Figure 2(A), for example. This is followed by an additional step of changing the sign of the postextending velocity, that is,

$$\mathbf{u}_g = -\mathbf{u}_{i+1,j}^{n+1}, \quad (24)$$

to fulfill the nonslip condition. In some cases, the slip velocity condition for inviscid flows is also in consideration, then we obtain the image velocity vector with respect to the interface using

$$\mathbf{u}_g = \mathbf{u}_{i+1,j}^{n+1} - 2(\mathbf{u}_{i+1,j}^{n+1} \cdot \mathbf{n})\mathbf{n}. \quad (25)$$

Regarding the above extending procedures, we emphasize that the present method merely requires (1) the shortest distance from each solid body cell to the fluid–solid interface to determine the narrow band of ghost cells and (2) the unit normal vector of each ghost cell to the fluid–solid interface. As the fluid–solid interface is usually defined by the mask function χ , a levelset function is widely used to play such a role. However, the levelset function involves a costly on-the-fly reinitialization procedure when the fluid–solid interface is moving. In this work, we can simply use a series of Lagrangian points (markers) to outline the closed fluid–solid interface in an arbitrary user-defined polygon shape,²⁷ so that geometry-related information including the shortest distance and the interface normal can be conveniently provided in an efficient and robust manner. As shown in Figure 2(B), the Lagrangian points are independent of the computational cells/points in the background grid and queued in the counter-clockwise direction. Given the ghost cell G inside the solid domain, the shortest distance from G to the interface is the shortest one of distances from G to edges $P_k P_{k+1}$ using a simple sorting algorithm. Note that an eligible distance candidate should satisfy

$$\overrightarrow{GP_k} \cdot \overrightarrow{P_k P_{k+1}} \leq 0, \quad \overrightarrow{GP_{k+1}} \cdot \overrightarrow{P_k P_{k+1}} \geq 0, \quad k = 1, \dots, np, \quad (26)$$

where np is the number of points/vertices of the polygon, and $P_{k+1} \leftarrow P_1$ when $k = np$. Besides, perpendicular to the nearest edge is the normal vector from cell G we desire. Other properties such as the centroid of the solid body, total mass, and inertial moment with respect to the centroid are all available if needed.

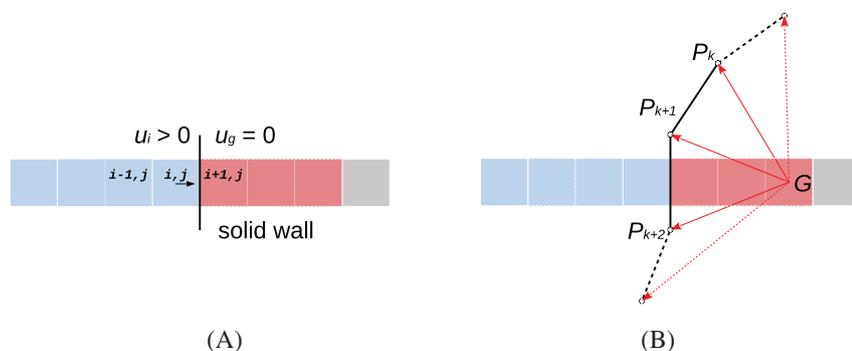


FIGURE 2 Schematics of (A) treating the nonslip condition in the penalization method and (B) the Lagrangian points connected polygon representing the fluid–solid interface

2.4 | Spatial and temporal discretization in fluid domain

Discretizing the compressible Navier–Stokes equations in Equation (2) on an adaptive Cartesian grid,²⁵ flow solver in the present study can conveniently employ high-order low-dissipation shock-capturing schemes to reconstruct the inviscid convective flux terms, as well as high-order central difference schemes to compute the viscous diffusive fluxes. In this study, the fifth-order WENO-LLF²⁸ finite difference scheme for multispecies reactive flows, based on characteristic decomposition²⁹ and upwind flux splitting, and a simple fourth-order central difference scheme are used, respectively. To balance the overall accuracy in time and the computational cost, the temporal integration utilizes the strong stability preserving second-order explicit Runge-Kutta scheme

$$\begin{aligned} U^* &= U^n + \Delta t \mathcal{L}(U^n), \\ U^{n+1} &= \frac{1}{2}U^n + \frac{1}{2}U^* + \frac{1}{2}\Delta t \mathcal{L}(U^*), \end{aligned} \quad (27)$$

for a timestep Δt constrained by the CFL condition.

Remark 3. Despite the present variable-extended IBM being more like a smooth interface method, its capability of sharp interface capturing can be alternatively strengthened by improving the spatial resolution near the interface with adaptive grid refinement.

2.5 | Fluid–solid interaction

In the scope of this article, we simply consider the stationary solid body embedded in the computational domain. Hence, the fluid–solid interaction is a one-way coupling here. That is to say the solid body prevents the flow from passing through directly but is not influenced by the fluid. However, further implementation of moving solid bodies and the related dynamics is straightforward. For example, given the flowfield information around the solid body, fluid–solid interaction can be considered by imposing forces such as pressure and viscous forces from the fluid upon the solid body surface. Each vertex of the solid body polygon can be used as a sampling point that measures the local pressure and viscous forces. The integrated forces and torques will then be used to determine the motion of rigid bodies if the rigid-body dynamics is considered, see more details in References 30,31.

3 | NUMERICAL RESULTS AND DISCUSSION

In this section, we attempt to validate the present variable-extended IBM using several numerical examples concerning simple single species (ideal air) or multispecies reactive flows with finite-rate nonequilibrium chemistry past the solid body obstacle in 1D or 2D space. For solving the ODE system of chemical kinetics, an efficient and robust explicit ODE solver, CHEMEQ2,³² is employed. The kinetic mechanism which describes the species and reactions will be specified in relevant examples.

3.1 | Case 1: One-dimensional acoustic wave reflection

To assess the effectiveness of the present variable-extended IBM in capturing the reflective nature of a solid wall, we use the reflection of a 1D acoustic wave at the interface between the fluid and a solid obstacle. The entire computational domain Ω is $[0,0.6]$. The fluid occupies $\Omega_f = [0, 0.5]$ and the solid occupies $\Omega_s = [0.5, 0.6]$ with the interface being located at $x = 0.5$. The initial conditions include the density, velocity and pressure perturbations of the Gaussian distribution

$$\rho' = u' = p' = \epsilon \exp \left[-\ln(2) \frac{(x - 0.25)^2}{0.004} \right], \quad (28)$$

where the wave amplitude is $\epsilon = 10^{-3}$, to be applied upon the spatially uniform field of $\rho = 1, u = 0, p = \frac{1}{\gamma}$, and $c = 1$ being the speed of sound. The fluid material is air with $\gamma = 1.4$. Theoretically, wave reflection at the wall $x = 0.5$ should be perfectly symmetric, and the reflected acoustic wave should have the same shape and size, only with opposite velocity.^{17,18} Hence, the pulse at $t = 0.5$ ought to recover its initial position. Computations consider the flow to be inviscid and are based on several uniform grids with the increasing number of grid cells of $N_c = 60, 120, 240, 480, 960,$ and 1920 , respectively. A very tiny timestep of $\Delta t = 1 \times 10^{-5}$ is used to minimize the temporal integration errors. For the penalization method, we set $\phi = 0.01$ as suggested in Reference 1 and we found the above timestep is sufficiently small for a simple explicit Euler scheme to solve the penalized density equation in Equation (19).

Figure 3 shows the computed profiles of the pressure perturbation at $t = 0.5$ as the acoustic wave should recover its initial position, centered at $x = 0.25$. It is shown that for three varying grids, the present variable-extended IBM exhibits perfectly agreeable results with the reference profile, not only in preserving the perturbation amplitude of ϵ but also in reducing the phase shift from pulse center $x = 0.25$. The penalization method, however, gives less accurate results, especially when the grid resolution is coarse below $N_c = 240$. In the penalization method, the spikes inside the solid domain ($x > 0.5$) are apparent, which is also the case in figure 3(B) of Reference 17. Both methods are able to produce converging solutions as the grid resolution improves. To evaluate the accuracy of both methods, we calculate the errors by comparing the computed wave profile with its initial counterpart within the fluid domain, namely, $x \in [0, 0.5]$. In Figure 4, L_1 and L_∞ error norms of pressure and velocity, respectively, are plotted. As it is shown, although all the error norms follow the first-order scaling with the increasing grid resolution, the present method produces obviously smaller absolute errors than the penalization method. Note that, for both methods, these errors are mostly attributed to how the reflective wall condition at the fluid–solid interface is treated.

3.2 | Case 2: Subsonic flow past a circular cylinder

A well-known benchmark test, that is, subsonic weakly compressible flow past a circular cylinder is used to validate the present IBM together with the adaptive Cartesian grid-based flow solver. The free stream is featured by $Re = 185$ and $Ma = 0.25$, such that in our computation the dimensional static pressure and temperature of the free stream are set to be 120 Pa and 300 K, respectively, with the cylinder diameter $D = 0.02823$ m. Size of the rectangular domain is $60D \times 40D$ with the circular cylinder center at $(20D, 20D)$. At $t = 0$, the computational grid is adaptively

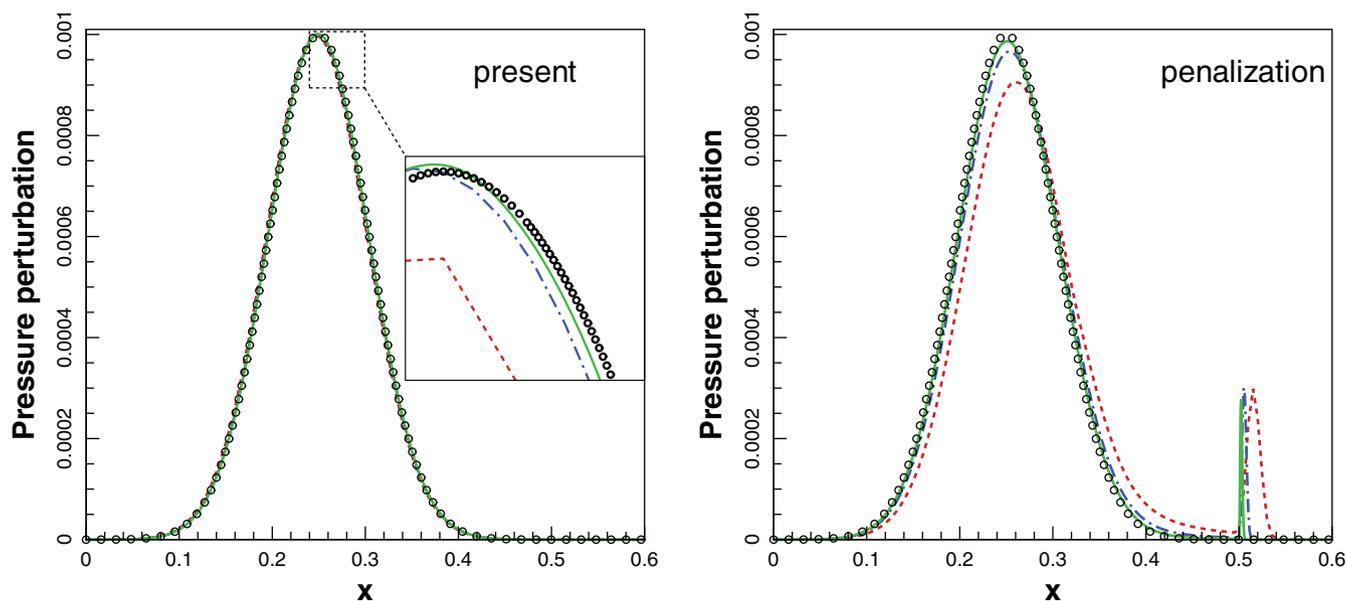


FIGURE 3 (Case 1) The computed pressure perturbations at $t = 0.5$ by the present variable-extended immersed boundary method and the penalization method, respectively. Red dashed lines: $N_c = 120$; Blue dash dot lines: $N_c = 480$; Green solid lines: $N_c = 1920$; Black circle symbols: the reference solution

FIGURE 4 (Case 1) L_1 and L_∞ error norms of computed pressure and velocity by variable-extended immersed boundary method and the penalization method, respectively

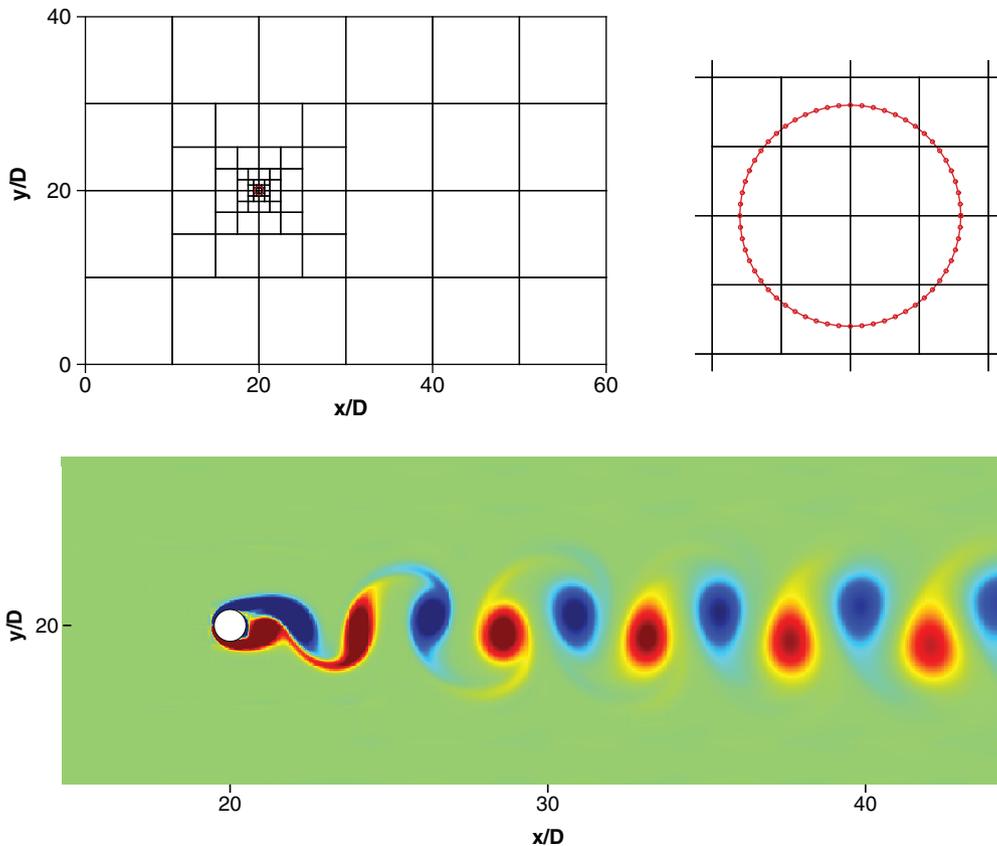
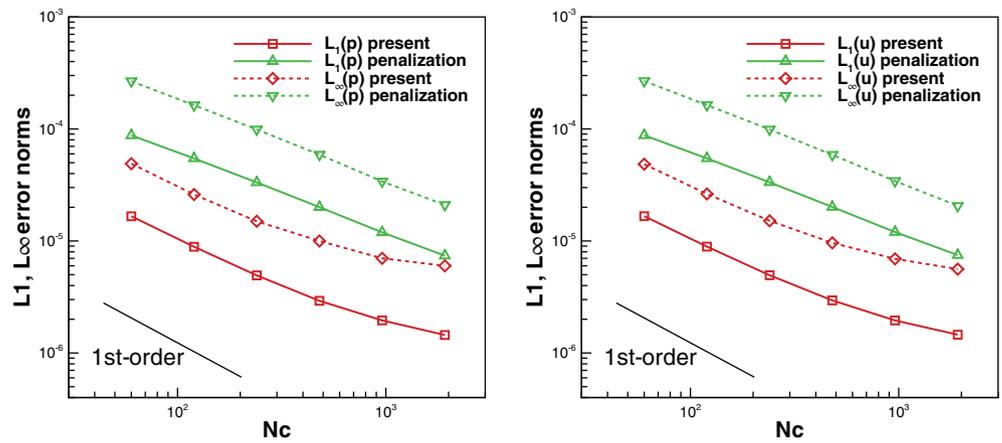


FIGURE 5 (Case 2) Computational domain including the flow domain and the solid circular cylinder, with the adaptively refined grid of blocks at $t^* = 0$ (top), and a snapshot of the spanwise vorticity field of evolutionary vortex shedding after the cylinder at $t^* = 98.35$, where $t^* = tu_\infty/D$ (bottom)

refined according to the inflow condition and is represented by blocks as shown in Figure 5, including the 61-point circular cylinder zoomed in aside. Note that each block contains a fixed array of 16×16 computational cells. Hence, the equivalent spatial resolution would be 1536×1024 over the entire domain, except the cylinder boundary with one level more refinement. The cylinder boundary is considered as an adiabatic and nonslip wall so that velocity inside the cylinder is treated by Equation (24). Other boundary conditions consist of the inflow condition at the left side and the outflow condition at the remaining three sides. A fixed CFL number of 0.8 is used. The entire computational domain at $t = 0$ is initialized using the freestream state except that the velocities inside the solid cylinder are zeros.

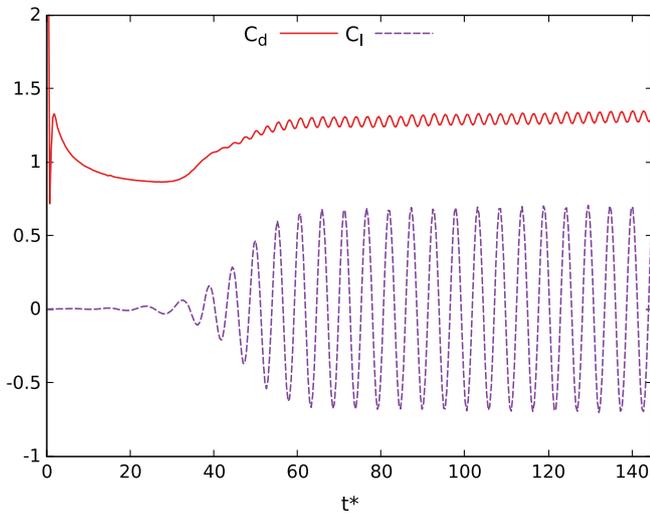


FIGURE 6 (Case 2) Time evolution of coefficients of the drag and lift forces, C_d and C_l

	$Re = 185$		
	$\overline{C_d}$	$C_{l(r.m.s)}$	St
Experimental results ³³	1.28	–	0.19
Lu and Dalton ³⁴	1.31	0.422	0.195
Guilmineau and Queutey ³³	1.287	0.443	0.195
Liu and Hu ³⁵	1.289	0.451	0.197
Khalili et al. ⁸	1.282	0.431	0.191
Present study	1.2894	0.4841	0.1907

TABLE 1 Comparison of computed data for Case 2 with available numerical and experimental results

In Figure 5 with the vorticity field at the bottom, we can clearly observe the nonsymmetric vortex shedding in the laminar wake behind the cylinder. Time evolutionary lift and drag coefficients C_l and C_d are plotted in Figure 6, indicating the unsteady vortex shedding tends to be regularly periodic at later stage. In Table 1, we quantitatively compare the present results, which are conveniently sampled and integrated using local quantities at the Lagrangian points of the cylinder, with other numerical and experimental results available. A very satisfactory agreement with reference results in the time-averaged drag coefficient $\overline{C_d}$ and the Strouhal number St can be seen, which implies the capacity of capturing transient behaviors in the flowfield by the present flow solver with the proposed variable-extended IBM. The amplitude of the lift coefficient C_l in Figure 6 and the resulted $C_{l(r.m.s)}$ are greater than the reference numerical results which also exhibit a considerably large variation of $C_{l(r.m.s)}$ between themselves. The reason might be attributed to the limited number of sampling points and the interpolation scheme from its surrounding grid cells to each sampling point in statistics.

3.3 | Case 3: Reactive gas mixture shock tube

We begin to simulate gaseous reactive flows by considering a 1D reactive shock tube to evaluate the performance of the present method compared with the penalization method and traditional “body-fitted” exact reflective wall boundary condition. A reactive gas mixture is placed in a closed tube, in which a shock hits the left-side solid wall boundary and reflects off. A reaction wave occurs at the boundary after a delay of induction, then catches up and merges with the right-moving shock wave to develop a detonation wave. The reactive mixture is characterized by a 2/1/7 molar ratio of $H_2/O_2/Ar$. Initial conditions in References 29,36,37 are

$$\begin{aligned}
 (\rho, u, p)_L &= (0.072 \text{ kg/m}^3, 0 \text{ m/s}, 7173 \text{ Pa}), \\
 (\rho, u, p)_R &= (0.18075 \text{ kg/m}^3, -487.34 \text{ m/s}, 35,594 \text{ Pa}),
 \end{aligned} \tag{29}$$

with the left and right states being separated in the middle of a 12 cm-long domain. In previous studies, exact wall boundary condition is often applied for the left end of tube using

$$(u, \rho, T, p)_g = (-u, \rho, T, p)_i, \quad (30)$$

where subscript “g” and “i” denote the ghost cells and their image internal cells, respectively. Instead, in the present IBM and the penalization method, a solid body is immersed within $0 \leq x \leq 3$ cm to function as the wall and the entire shock tube is translated by 3 cm rightwards. Outflow condition for the right end of tube uses the first-order extrapolation. A 9-species 23-reaction hydrogen/air combustion mechanism³⁸ is considered here. Results are analyzed at $t = 210 \mu\text{s}$ with $N_c = 200, 400,$ and 800 in the fluid domain, respectively, and a fixed CFL number of 0.75. The solid domain, if involved, uses equal grid spacings with the fluid domain.

In Figure 7, it can be seen that using the penalized nonslip velocity condition results in slower propagation of the detonation wave at all the three grids. We have attributed such inaccuracy to the loss of total energy at

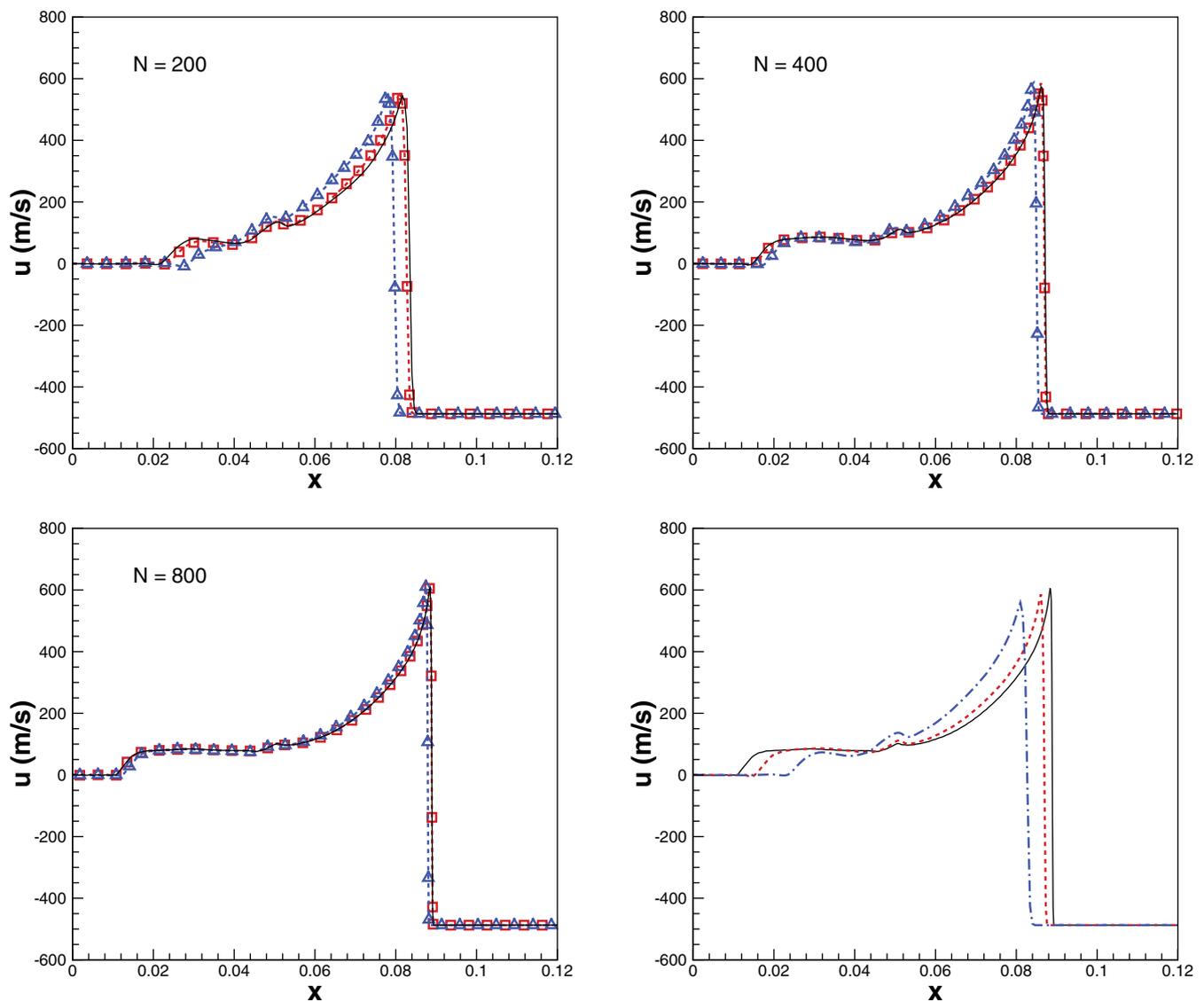


FIGURE 7 (Case 3) Reactive shock tube velocity profiles at $t = 210 \mu\text{s}$ using 200, 400, and 800 grid cells, respectively. Note that all the profiles of the present IBM and the penalization method are translated leftwards by 3 cm. Dashed lines with red square symbols: the present IBM; dash dot lines with blue triangle symbols: the penalization method; solid black lines: exact wall boundary condition. The fourth subfigure shows the convergence of the present IBM with increasing the spatial resolution. IBM, immersed boundary method

the wall by the penalization method, as mentioned in previous section. By contrast, the present variable-extended method yields much agreeable detonation waves in terms of the location of the reacting front, the velocity peak value and the overall profile of the reflected waves, against the exact wall boundary condition solution. Grid convergence of the present method is especially shown to validate the numerical accuracy of coping with unsteady reactive flows.

3.4 | Case 4: Shock-induced detonation of Lehr's experiments

We continue to simulate the gaseous reactive flow from Lehr's benchmark experiments,³⁹ by considering a spherical projectile with radius $r = 7.5$ mm flying past the premixed reactive gas mixture of $\text{H}_2/\text{O}_2/\text{N}_2$ in three scenarios:

- (1) *super-detonation case*.^{40,41} The surrounding reactive gas mixture is featured by the molar ratio of $\text{H}_2:\text{O}_2:\text{N}_2$ being 2:1:3.76, static pressure of $p_\infty = 320$ mmHg and static temperature of $T_\infty = 293$ K. The projectile velocity is $u_\infty = 2605$ m/s, corresponding to a supersonic flight Mach number of $M = 6.46$.
- (2) *subdetonation case*.⁴⁰ The surrounding reactive gas mixture is featured by the molar ratio of $\text{H}_2:\text{O}_2:\text{N}_2$ being 2:1:0, static pressure of $p_\infty = 186$ mmHg and static temperature of $T_\infty = 293$ K. The projectile velocity is $u_\infty = 1892$ m/s, corresponding to $M = 3.55$.
- (3) *trans-detonation case*.⁴¹⁻⁴⁴ The surrounding reactive gas mixture is the same as that of the superdetonation case. The projectile velocity is lower, that is, $u_\infty = 1804$ m/s, corresponding to $M = 4.48$.

At $t = 0$, the computational grid is adaptively refined according to the initial condition and is represented by blocks as shown in Figure 8. Freestream inflow condition is set for the left boundary of the fluid domain. Boundary conditions are easily imposed for supersonic flows by using the first-order extrapolation. For the bottom side, axisymmetric boundary condition should be applied and additional source terms accounting for the 2D axisymmetric model are given by

$$S(U) = -\frac{v}{y}(\rho, \rho u, \rho v, (\rho e_t + p), \rho y_1, \dots, \rho y_{ns-1})^T. \quad (31)$$

The spherical projectile surface is considered as an adiabatic wall so that Neumann boundary condition for temperature is applied. As the effect of transport properties is known to be negligible,⁴² concerned computations are performed on a basis of the inviscid assumption. A fixed CFL number of 0.75 is used.

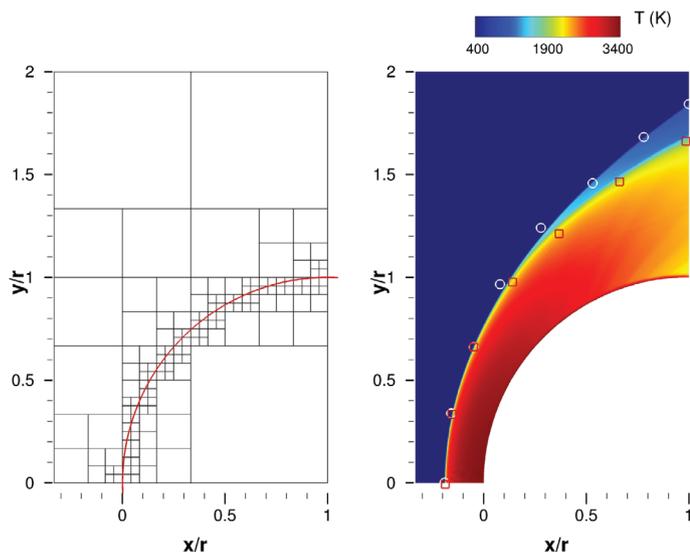


FIGURE 8 (Case 4) Computational domain including the flow domain and the solid spherical projectile, with the adaptively refined grid of blocks at $t = 0$ (left) (note that for the superdetonation case, each block contains 16×16 grid cells), and the computed steady superdetonation temperature field at $t = 4 \times 10^{-5}$ s (right), upon which experimental measurements of the shock and combustion front are marked by red circle symbols and black square symbols, respectively. The Evans' hydrogen/air combustion mechanism (seven species/eight reactions) is used⁴⁵

Considering the superdetonation first, in Figure 8, the right figure depicts the fully developed steady-state temperature distribution around the spherical head. We can see that near the stagnation streamline, the reacting front of the superdetonation merges with the detached bow shock, because the flow past the shock decelerates so quickly that the high temperature and pressure environment would lead to immediate ignition of the reactive mixture. In the circular direction upwards, bifurcation between the reacting front and the bow shock arises since the induction time/length of ignition tends to be longer. Experimental measurement of the two discontinuity positions is marked upon the present numerical result, matching excellently with each other.

The subdetonation case is described by the computed steady temperature distribution in Figure 9. As it is shown, the reacting zone close to the spherical head is much thinner than that of the superdetonation case, being apart from the detached shock. In comparison with the experimental shadowgraph, it is readily to see that the present numerical result including the shock and reacting front positions and curves, no matter ahead of the projectile or away from the projectile, agrees well with its reference.

The transdetonation case involves an interesting phenomenon of the oscillating reacting front. As shown in Figure 10 (top), the instantaneous temperature fields exhibit obvious unsteady combustion with a noncircular reacting front, which belongs to the regular regime⁴⁴ in that the flow oscillates with high frequency and low amplitude. Of all the Lehr's experiments, the Mach 4.48 case gives a measured frequency of 425 kHz. In the present computation, by analyzing the long-time history of pressure at the stagnation point in Figure 10 (bottom left), the calculated frequency of regular cycles by DFT analysis is about 395 kHz for the computation with a block size of 32×4 cells. By increasing the y -direction spatial resolution with a block size of 32×8 cells, the calculated frequency increases to 410 kHz. Other studies^{43,44} also suggest an improved numerical accuracy of the oscillation frequency by increasing the grid resolution. Back to Figure 10 (top), it is also interesting to observe that 1) a small strip of isolated combustion area forms in front of the main detonation wave at $t = 57.79 \mu\text{s}$, 2) it then develops upwards and merges with the main combustion area at $t = 58.75 \mu\text{s}$ and 3) the merged reacting front continues to move along the circular direction at $t = 59.71 \mu\text{s}$. In this process, we can also find that the thickness of the main combustion area along the stagnation streamline increases at the beginning to reach a peak value and then decreases. These three stages, from Figure 10 (bottom right), in fact correspond to the minimum, approximately the maximum and the mean value, respectively, of the oscillating pressure at the stagnation point in a cycle. As a result, the pressure oscillation at the stagnation point is believed to be highly related to the repeated stretch and shrink of the combustion area attached to the projectile head, which challenges the accuracy and robustness of the numerical method in use and verifies the present variable-extended IBM.

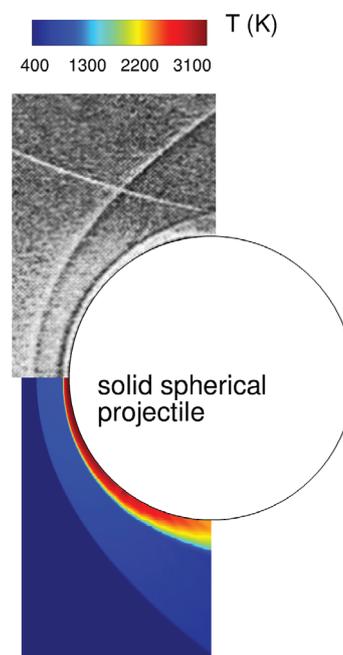


FIGURE 9 (Case 4) Comparison of the computed steady subdetonation temperature field (mirrored against the x -axis) with the experimental shadowgraph.^{39,40} Note that for the subdetonation case, each block contains 16×4 grid cells to reduce computational costs. The Jachimowski's hydrogen/air combustion mechanism (9 species/19 reactions) is used⁴²

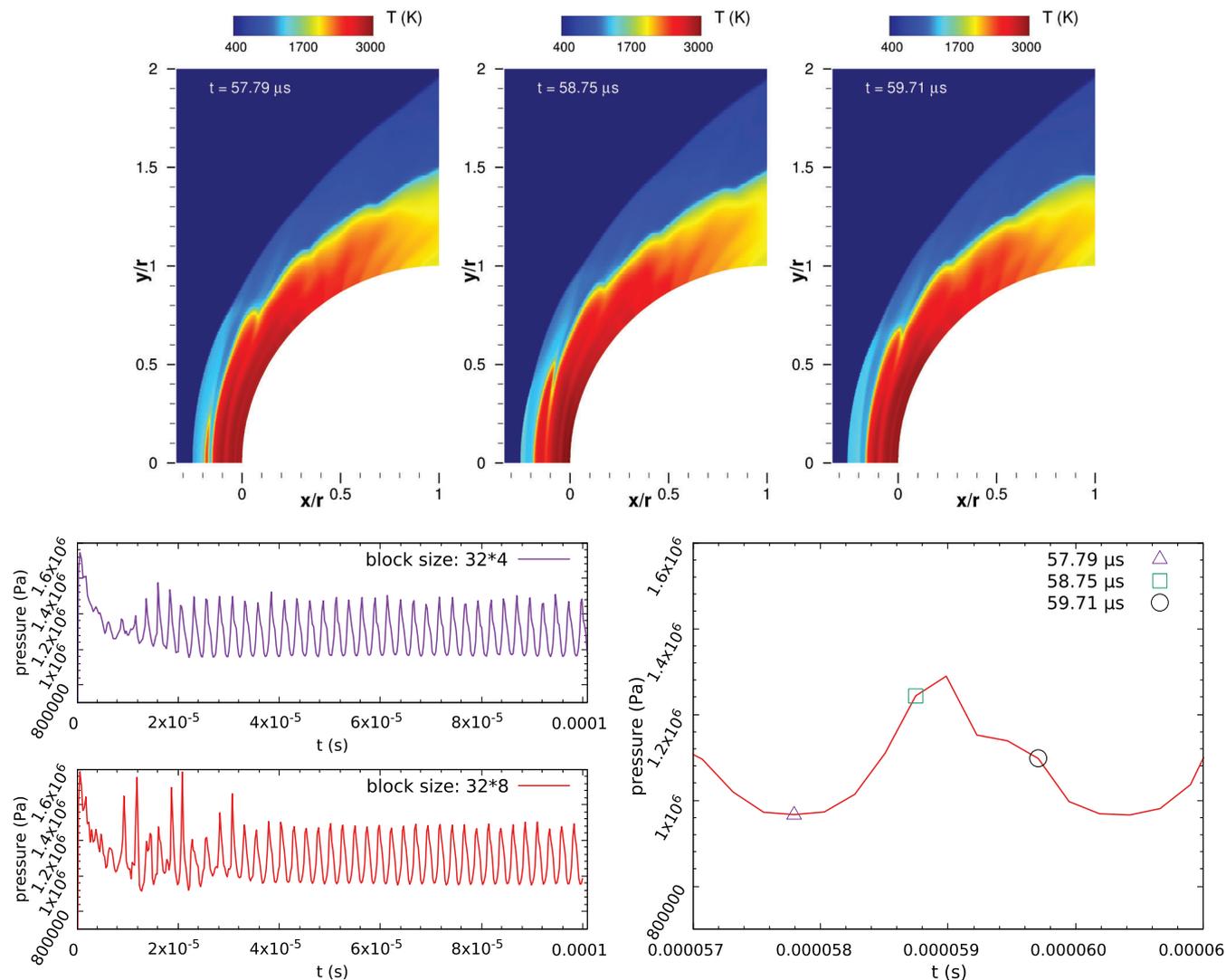


FIGURE 10 (Case 4) The unsteady transdetonation temperature fields at three time instants (top), the histories of pressure at the stagnation point over time using two grids of blocks (bottom left) and the zoomed-in pressure oscillation cycle (bottom right). The Jachimowski's mechanism (9 species/19 reactions) is also used here

4 | CONCLUSIONS

In this article, a variable-extended IBM has been proposed to simulate compressible gaseous reactive flows past solid bodies. In order to alleviate the numerical stiffness of the penalized continuity equation as well as to facilitate the implementation of the Neumann boundary condition in BPM, an extending step from the fluid to solid domain within a narrow band is developed for relevant primitive variables including but not limited to velocity, pressure and temperature. With the help of an EoS, density of ghost cells inside the solid domain can be determined using the extended steady-state pressure and temperature, instead of solving the stiff penalized continuity equation directly. The adiabatic nonslip or slip wall condition can be thereby easily imposed. Meanwhile, the advantage of the BPM in avoiding the local geometric complexity of the solid-fluid interface is maintained, without considering the cut cell types, intersection points and image points. Combined with an adaptive Cartesian grid-based flow solver for multispecies compressible Navier–Stokes equations with finite-rate nonequilibrium chemistry, the present variable-extended IBM has been tested by 1D and 2D numerical examples, and exhibits satisfactory performance in capturing (1) the reflective nature of fluid–solid interface and (2) the nonreactive or reactive, weakly compressible or high Mach number, steady or unsteady flow-fields in a simple and robust manner. Besides, treating the velocity carefully in the ghost cells inside the solid domain helps to conserve the total energy and results in more accurate propagation of the detonation front in energy-sensitive

reactive flows. Some future work concerning the solid body dynamics and more complicated multiphase flows can be conducted.

ACKNOWLEDGMENT

The financial support from the EU Marie Skłodowska-Curie Innovative Training Networks (ITN-ETN) (Project ID: 675528-IPPAD-H2020-MSCA-ITN-2015) for the first author is gratefully acknowledged. Open access funding enabled and organized by Projekt DEAL.

DATA AVAILABILITY STATEMENT

Some or all data, models, or code generated or used in this study are available from the corresponding author by request. (List items).

ORCID

Jian-Hang Wang  <https://orcid.org/0000-0002-1364-8389>

REFERENCES

1. Piquet A, Roussel O, Hadjadj A. A comparative study of brinkman penalization and direct-forcing immersed boundary methods for compressible viscous flows. *Comput Fluids*. 2016;136:272-284.
2. Tseng YH, Ferziger JH. A ghost-cell immersed boundary method for flow in complex geometry. *J Comput Phys*. 2003;192(2):593-623.
3. Ghias R, Mittal R, Dong H. A sharp interface immersed boundary method for compressible viscous flows. *J Comput Phys*. 2007;225(1):528-553.
4. Mittal R, Dong H, Bozkurttas M, Najjar FM, Vargas A, Loebbecke AV. A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *J Comput Phys*. 2008;227(10):4825-4852.
5. Seo JH, Mittal R. A high-order immersed boundary method for acoustic wave scattering and low-mach number flow-induced sound in complex geometries. *J Comput Phys*. 2011;230(4):1000-1019.
6. Lakshminarayan V, Farhat C, Main A. An embedded boundary framework for compressible turbulent flow and fluid-structure computations on structured and unstructured grids. *Int J Numer Methods Fluids*. 2014;76(6):366-395.
7. Brehm C, Hader C, Fasel HF. A locally stabilized immersed boundary method for the compressible Navier-stokes equations. *J Comput Phys*. 2015;295:475-504.
8. Khalili ME, Larsson M, Muller B. Immersed boundary method for viscous compressible flows around moving bodies. *Comput Fluids*. 2018;170:77-92.
9. Mittal R, Iaccarino G. Immersed boundary methods. *Annu Rev Fluid Mech*. 2005;37:239-261.
10. Clarke DK, Hassan HA, Salas MD. Euler calculations for multielement airfoils using Cartesian grids. *AIAA J*. 1986;24(3):353-358.
11. Schneiders L, Günther C, Meinke M, Schröder W. An efficient conservative cut-cell method for rigid bodies interacting with viscous compressible flows. *J Comput Phys*. 2016;311:62-86.
12. Colella P, Graves DT, Keen BJ, Modiano D. A cartesian grid embedded boundary method for hyperbolic conservation laws. *J Comput Phys*. 2006;211(1):347-366.
13. Forrer H, Jeltsch R. A higher-order boundary treatment for Cartesian-grid methods. *J Comput Phys*. 1998;140(2):259-277.
14. Boiron O, Chiavassa G, Donat R. A high-resolution penalization method for large mach number flows in the presence of obstacles. *Comput Fluids*. 2009;38(3):703-714.
15. Arquis E, Caltagirone J. Sur les conditions hydrodynamiques au voisinage d'une interface milieu Fluide-Milieu Poreux: application à la convection naturelle. *C.R Acad Sci Paris II*. 1984;299:1-4.
16. Angot P, Bruneau CH, Fabrie P. A penalization method to take into account obstacles in incompressible viscous flows. *Numer Math*. 1999;81(4):497-520.
17. Liu Q, Vasilyev OV. A brinkman penalization method for compressible flows in complex geometries. *J Comput Phys*. 2007;227(2):946-966.
18. Anand N, Pour NE, Klimach H, Roller SP. Utilization of the brinkman penalization to represent geometries in a high-order discontinuous Galerkin scheme on octree meshes. *Symmetry*. 2019;11(9):1126.
19. Browndymkoski E, Kasimov N, Vasilyev OV. A characteristic based volume penalization method for general evolution problems applied to compressible viscous flows. *J Comput Phys*. 2014;262:344-357.
20. Hardy B, Wilde JD, Winckelmans G. A penalization method for the simulation of weakly compressible reacting gas-particle flows with general boundary conditions. *Comput Fluids*. 2019;190:294-307.
21. Wilke CR. A viscosity equation for gas mixtures. *J Chem Phys*. 1950;18(4):517-519.
22. Feireisl E, Neustupa J, Stebel J. Convergence of a Brinkman-type penalization for compressible fluid flows. *J Differ Equ*. 2011;250(1):596-606.
23. Strang G. On the construction and comparison of difference schemes. *SIAM J Numer Anal*. 1968;5(3):506-517.
24. Yoon S, Jameson A. Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-stokes equations. *AIAA J*. 1988;26(9):1025-1026.

25. Han LH, Hu XY, Adams NA. Adaptive multi-resolution method for compressible multi-phase flows with sharp interface model and pyramid data structure. *J Comput Phys*. 2014;262:131-152.
26. Fu L, Hu XY, Adams NA. Single-step reinitialization and extending algorithms for level-set based multi-phase flow simulations. *Comput Phys Commun*. 2017;221:63-80.
27. Radouan B, Bouvelle B, Bouali Z, Mura A. An immersed boundary method to simulate compressible reactive flows featuring shock-wave interactions with three-dimensional solid obstacles. *Paper presented at: 25th ICDEERS, August 2-7, 2015 Leeds, UK*; 2015.
28. Chi-Shu W.. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, Lecture Notes in Mathematics*. Vol 1697. Cetraro, Italy: Springer; 1997:325.
29. Wang J-H, Pan S, Hu XY, Adams NA. Partial characteristic decomposition for multi-species Euler equations. *Comput Fluids*. 2019;181:364-382.
30. Kim Y, Peskin CS. A penalty immersed boundary method for a rigid body in fluid. *Phys Fluids*. 2016;28(3):033603.
31. Bergmann M, Iollo A. Modeling and simulation of fish-like swimming. *J Comput Phys*. 2011;230(2):329-348.
32. Mott DR, Oran ES. *Chemeq2: A Solver for the Stiff Ordinary Differential Equations of Chemical Kinetics. Technical Report*. Washington, DC: Naval Research Lab; 2001.
33. Guilmineau E, Queutey P. A numerical simulation of vortex shedding from an oscillating circular cylinder. *J Fluids Struct*. 2002;16(6):773-794.
34. Lu X, Dalton C. Calculation of the timing of vortex formation from an oscillating cylinder. *J Fluids Struct*. 1996;10(5):527-541.
35. Liu C, Hu C. An efficient immersed boundary treatment for complex moving object. *J Comput Phys*. 2014;274:654-680.
36. Fedkiw R, Merriman B, Osher S. High accuracy numerical methods for thermally perfect gas flows with chemistry. *J Comput Phys*. 1997;132(2):175-190.
37. Ferrer PJM, Buttay R, Lehnasch G, Mura A. A detailed verification procedure for compressible reactive multicomponent navier–stokes solvers. *Comput Fluids*. 2014;89(89):88-110.
38. Conaire MO, Curran HJ, Simmie JM, Pitz WJ, Westbrook CK. A comprehensive modeling study of hydrogen oxidation. *In J Chem Kinet*. 2004;36(11):603-622.
39. Lehr HF. Experiments on shock-induced combustion. *Astronautica Acta*. 1972;17(4):589-597.
40. Li Y, Tao T. Resolving the shock-induced combustion by an adaptive mesh redistribution method. *J Comput Phys*. 2007;224(2):587-600.
41. Ess PR, Allen CB. Parallel computation of two-dimensional laminar inert and chemically reactive multi-species gas flows. *Int J Numer Methods Heat Fluid Flow*. 2005;15(3):228-256.
42. Choi J, Jeung I, Yoon Y. Computational fluid dynamics algorithms for unsteady shock-induced combustion, Part 1: validation. *AIAA J*. 2000;38(7):1179-1187.
43. Kumar PP, Kim K, Oh S, Choi J. Numerical comparison of hydrogen-air reaction mechanisms for unsteady shock-induced combustion applications. *J Mech Sci Technol*. 2015;29(3):893-898.
44. Pavalavanni PK, Sohn CH, Lee BJ, Choi J-Y. Revisiting unsteady shock-induced combustion with modern analysis techniques. *Proc Combust Inst*. 2019;37(3):3637-3644.
45. Evans JS, Schexnayder CJ. Influence of chemical kinetics and unmixedness on burning in supersonic hydrogen flames. *AIAA J*. 1980;18(2):188-193.

How to cite this article: Wang J-H, Zhang C. The variable-extended immersed boundary method for compressible gaseous reactive flows past solid bodies. *Int J Numer Methods Eng*. 2021;122:2221–2238. <https://doi.org/10.1002/nme.6619>