On the use of high order central difference schemes for differential equation based wall distance computations

Hemanth Chandra Vamsi Kakumani[a], Nagabhushana Rao Vadlamani[a,*], Paul Gary Tucker[b]

[a]*Department of Aerospace Engineering, Indian Institute of Technology Madras, Chennai, 600036, India*

[b]*Department of Engineering, University of Cambridge, Cambridge, CB2 1PZ, UK*

Abstract

A computationally efficient high-order solver is developed to compute the wall distances by solving the relevant partial differential equations, namely: Eikonal, Hamilton-Jacobi (HJ) and Poisson equations. In contrast to the upwind schemes widely used in the literature, we explore the suitability of high-order central difference schemes (explicit/compact) for the wall-distance computation. While solving the Hamilton-Jacobi equation, the high-order central difference schemes performed approximately $1.4 - 2.8$ times faster than the upwind schemes with a marginal improvement in the solution accuracy. A new pseudo HJ formulation based on the localized artificial diffusivity (LAD) approach has been proposed. It is demonstrated to predict results with an accuracy comparable to that of the Eikonal equation and the simulations are ≈ 1.5 times faster than the baseline HJ solver using upwind schemes. A curvature correction is also incorporated in the HJ equation to correct for the near-wall errors due to concave/convex wall curvatures. We demonstrate the efficacy of the proposed methods on both the steady and unsteady test cases and exploit the unsteady wall-distance solver to estimate the instantaneous shape and burning surface area of a dendrite propellant grain in a solid propellant rocket motor.

Keywords: Wall distance, High order schemes, Hamilton-Jacobi equation, Localized Artificial Diffusivity (LAD).

1. Introduction

Nearest distance to the solid no-slip wall, referred to as wall-distance in the literature, is often used in the formulation of several eddy-resolving or turbulence modelling strategies. For example, original formulation of the Detached Eddy Simulations (DES) uses wall distance to switch between the modelled and resolved regions of turbulence. Reynolds Averaged Navier Stokes (RANS) [12] models like Spalart Allmaras, $k - \epsilon$, $k - \omega$ use wall-distance for the near-wall treatment. Such strategies can also be used to alter the dissipation term in the turbulence models or to introduce surface roughness effects by locally altering the length scales of flow using wall-distance [8, 10]. Wall-distances are also used in the peripheral flow applications such as multiphase flows. Accuracy of the wall-distance values and their efficient computation is important in these fluid flow simulations. In addition to Computational Fluid Dynamics (CFD), wall-distances are also encountered in other applications such as Computer Aided Design (CAD) and automated meshing.

There are various search-based algorithms available in the literature to estimate the wall-distances [13, 14]. However, search algorithms are computationally expensive and suffer from weak scalability, specifically on large meshes and with increasing number of wall faces in the computational domain. Search algorithms

---

*Corresponding author: `nrv@iitm.ac.in` (Nagabhushana Rao Vadlamani)

also produce inaccurate solutions on non-orthogonal and unstructured grids due to Eucludian and projection based measurement approaches they employ. Search approaches can sometimes produce non-smooth or heavily discontinuous solutions, particularly on low-quality meshes which are non-orthogonal and highly skewed. These inaccuracies can lead to convergence issues of the flow simulations.

On the other hand, differential equation based wall-distance strategies [2, 3, 4] overcome most of the limitations encountered by the search-based algorithms. These approaches are highly scalable, much more accurate than search-based algorithms and produce smoother solutions in the computational domain. Three classes of differential equation based wall-distance approach include: Eikonal, Hamilton-Jacobi (H-J) and Poisson equations, which will be described in detail in the subsequent section. The formulations of these equations is similar to those of the differential equations governing the flow. Hence, the wall-distance differential equations can be readily implemented into the CFD solvers taking advantage of the existing functions/subroutines of the code that are already written. In addition, the differential equation based approach is demonstrated to be robust even on poor quality grids by Tucker et al. [2]. Eikonal and H-J equations can be categorized as advection and advection-diffusion equations respectively. Tucker et al. [2] has proposed upwind schemes to drive the wall-distance equations to a steady state solution. Although upwind schemes are robust and accurate particularly while solving the Eikonal equation, it involves the repeated computation of the wave front direction for every iteration. This results in an increase in the CPU time per iteration and consequently the overall convergence time particularly on unstructured grids. The implementation of upwind scheme is also relatively elaborate in curvilinear coordinates when compared to the standard explicit finite difference or compact schemes.

In the context of high-fidelity eddy resolving simulations like Large Eddy Simulations (LES) or Direct Numerical Simulations (DNS), high-order schemes have been gaining increasing popularity as these schemes can accurately represent the high spatio-temporal frequencies in the transitional/turbulent flows at a substantially lower grid count than the low-order schemes. To accurately capture turbulence, non-dissipative high-order explicit and compact central schemes are preferred over the dissipative up-wind schemes. This leads to the multi-fold objectives of the current work. Firstly, we explore the suitability of using the high-order central difference schemes (explicit and compact schemes up-to 6th order accuracy) to solve the differential equations governing the wall distance. We compare the accuracy and computational efficacy of these schemes with the standard upwind schemes. Subsequently, a modified local artificial diffusivity (LAD) based Hamilton-Jacobi formulation will be introduced, which yields wall-distance estimates as accurate as the Eikonal equation with enhanced stability characteristics. To account for the loss of wall distance accuracy near the regions of high wall curvatures, we introduce a curvature correction to the HJ equation inline with that proposed by Nakanishi [22]. In contrast to the fixed wall test cases typically reported in the literature, we also demonstrate the efficacy of the current formulation on the unsteady moving bodies where the wall-distance changes with time, including the practical test case of the burnback of a propellant grain in a solid rocket motor. It should be noted that the scope of the paper is confined to demonstrating the suitability (both in terms of computational cost and accuracy) of the high-order methods towards computing the wall-distances. Exploring the effects of the aforementioned strategies on the flow-field predictions is beyond the scope of the current study.

Here is the brief outline of the paper. Section 2 describes the three classes of differential equations governing the wall distances. In Section 3, we present the numerical methodology employed for the spatial discretization, filtering and convergence of the baseline and enhanced solvers. The details of the test cases are given in section 4. In section 5, we present the results for several steady and unsteady test cases using different discretization schemes. We also demonstrate the performance of the proposed modifications (HJ-LAD/Curvature corrections) on both the computational cost and accuracy. We conclude the paper in section 6 summarizing the key findings of the study.

2. Governing Equations

2.1. Eikonal equation

The exact governing equation for wall-distance (ϕ) is the Eikonal equation (Eqn.1), which is a non-linear hyperbolic partial differential equation. It's formulation originates from the fact that the magnitude of wall distance gradient at all the points in the domain equals one [9]. The equation reads as follows.

$$|\nabla\phi| = 1 \quad or \ (\nabla\phi)^2 = 1 \quad or \ \mathbf{U} \cdot \nabla\phi = 1 \tag{1}$$

$$U_x\frac{\partial\phi}{\partial x} + U_y\frac{\partial\phi}{\partial y} + U_z\frac{\partial\phi}{\partial z} = 1 \tag{2}$$

where $\mathbf{U}$ in the Eqn.1 is defined as $\nabla\phi$, the Eikonal front propagation velocity. The equation can also be rewritten in its expanded form, Eqn.2, with U_x, U_y and U_z as the components of $\mathbf{U}$. Given the non-linearity of this equation, upwind schemes are generally used in the literature. Solving it using non dissipative central schemes is challenging as the high-frequency Gibbs oscillations across the sharp discontinuity of the Eikonal front leads to the divergence of the solution. Dissipation of some form is required to suppress these high frequency oscillations thereby enhancing the stability and ensuring robust convergence. Methods to solve this equation will be described in section 3.

2.2. Poisson equation

Spalding [21] proposed a numerically simplistic Poisson based method to compute wall distances. The governing equation is given by:

$$\nabla^2\phi' = -1 \tag{3}$$

$$\phi = \pm\sqrt{\sum_{j=1,3}\left(\frac{\partial\phi'}{\partial x_j}\right)^2} + \sqrt{\sum_{j=1,3}\left(\frac{\partial\phi'}{\partial x_j}\right)^2 + 2\phi'} \tag{4}$$

It should be noted that the approach involves solving for ϕ' as the variable in the Poisson equation (Eqn.3). The actual wall distance, ϕ, is subsequently computed using the auxiliary equation, Eqn.4. This approach only yields accurate wall distances close to the walls. In most of the turbulence modelling strategies, it is sufficient enough to estimate accurate wall-distances close to the walls where the viscous effects are dominant. Nevertheless, certain applications like Computer Aided Design (CAD), computing minimal surfaces, automated meshing [16] etc demand accurate wall distance away from the walls.

2.3. Hamilton-Jacobi (H-J) equation

Hamilton-Jacobi (HJ) equation, proposed by Tucker et al. [1], can be interpreted as a hybrid between the Eikonal and Poisson equations. With the advection term on the LHS and a Laplacian term on the RHS, HJ equation takes the general form Eqn 5.

$$\mathbf{U} \cdot \nabla\phi = 1 + \Gamma\nabla^2\phi \tag{5}$$

While the hyperbolic nature of advection term helps in algorithmic efficiency, the Laplacian operator on the right provides the required dissipation to suppresses high frequency Gibbs oscillations originating from the advection term. The equation hence offers a good alternative to the Eikonal equation due to its superior stability and robustness. Although the Laplacian operator enhances stability, the accuracy of the wall-distance estimates can deteriorate depending upon the formulation used for the term Γ.

2.3.1. Standard formulation for Γ

The formulation for Γ is generally modelled as a linear function of ϕ in the literature [1, 2] as shown in Eqn.6. Other non-linear variations (quadratic, cubic, logarithmic etc.) can also be used instead of Eqn.6.

$$\Gamma = \epsilon\phi \tag{6}$$

Here, ϵ is an arbitrary constant typically ranging between 0.1 and 1, chosen based on the mesh and geometry of the simulation. It is evident that Γ is zero near the walls where $\phi << 0$ (essentially solving the Eikonal equation) and progressively increases away from the wall. The HJ equation using this formulation is referred to as 'standard H-J equation' in the current work. It is worth noting that this Laplacian operator, in the original formulation by Tucker et al. [2], has been deliberately tuned for turbulence modelling. Near fine convex features (thin wires) or convex corners, the model exaggerates the wall-distance, $\phi > d_{true}$, to remedy the excessive turbulence dissipation. On the other hand, through $\phi < d_{true}$ at a concave corner, the Laplacian term naturally accounts for the damping effects due to extra walls.

2.3.2. New Localized Artificial Diffusivity (LAD) based formulation for Γ

We propose an adaptive formulation for Γ based on the Localized Artificial Diffusivity (LAD) approach to improve the accuracy. The formulation is inline with the LAD based shock capturing method proposed by Kawai and Lele [20]. At each time step, the dispersion errors originating from the non-linear terms in the governing equation must be damped. We use an adaptive value of Γ (Eqn. 7), instead of the linear variation (Eqn. 6). It should be noted that Γ is proportional to the fourth derivative of ϕ, which is useful in identifying the localized regions where the dispersion error is dominant. The new formulation introduces a desired amount of dissipation to suppress dispersion thereby reducing the overall error in the final solution.

$$\Gamma = \text{MIN}\left(\epsilon\phi, C\left|\sum_{l=1}^{3}\left[\sum_{m=1}^{3}\left(\frac{\partial\xi_l}{\partial x_m}\right)^2\right]^2\frac{\partial^4\phi}{\partial\xi_l^4}\Delta x_l^3\right|\right) \tag{7}$$

Here, ξ_l refers to generalized coordinates ξ, η and ζ. C is an arbitrary constant which is taken as 0.1 for all the cases reported in this study. With a 10% variation in C, the global errors are found to vary within 1%.

3. Computational Set-up

In this section, we provide details of the transformed governing equations in curvilinear coordinates, spatial discretization and filtering schemes, and the algorithms employed to solve these equations in the baseline and enhanced codes.

3.1. Governing equations in general coordinates

The current solver is written in generalized coordinate system with ξ, η, and ζ being the coordinate axis directions. Equations 2, 3 and 5 in Cartesian coordinates are transformed to the curvilinear coordinates as given by the equations 8, 9 and 10 respectively. All the equations were non-dimensionalized with the reference length scale L_{ref} and time scale τ.

Eikonal equation:

$$\hat{U}\frac{\partial\phi}{\partial\xi} + \hat{V}\frac{\partial\phi}{\partial\eta} + \hat{W}\frac{\partial\phi}{\partial\zeta} = 1 \tag{8}$$

4

Hamilton-Jacobi equation:

$$\hat{U}\frac{\partial\phi}{\partial\xi} + \hat{V}\frac{\partial\phi}{\partial\eta} + \hat{W}\frac{\partial\phi}{\partial\zeta} = 1 + \Gamma\left(\xi_x\frac{\partial U_x}{\partial\xi} + \eta_x\frac{\partial U_x}{\partial\eta} + \zeta_x\frac{\partial U_x}{\partial\zeta} + \xi_y\frac{\partial U_y}{\partial\xi} + \eta_y\frac{\partial U_y}{\partial\eta} + \zeta_y\frac{\partial U_y}{\partial\zeta} + \xi_z\frac{\partial U_z}{\partial\xi} + \eta_z\frac{\partial U_z}{\partial\eta} + \zeta_z\frac{\partial U_z}{\partial\zeta}\right)$$

(9)

where, $\hat{U}, \hat{V}$ and $\hat{W}$ are contravariant velocities. U_x, U_y and U_z are the components of Eikonal front propagation velocity defined as follows:

$$\hat{U} = \xi_x U_x + \xi_y U_y + \xi_z U_z$$
$$\hat{V} = \eta_x U_x + \eta_y U_y + \eta_z U_z$$
$$\hat{W} = \zeta_x U_x + \zeta_y U_y + \zeta_z U_z$$

$$U_x = \frac{\partial\phi}{\partial x} = \xi_x\frac{\partial\phi}{\partial\xi} + \eta_x\frac{\partial\phi}{\partial\eta} + \zeta_x\frac{\partial\phi}{\partial\zeta}$$

$$U_y = \frac{\partial\phi}{\partial y} = \xi_y\frac{\partial\phi}{\partial\xi} + \eta_y\frac{\partial\phi}{\partial\eta} + \zeta_y\frac{\partial\phi}{\partial\zeta}$$

$$U_z = \frac{\partial\phi}{\partial z} = \xi_z\frac{\partial\phi}{\partial\xi} + \eta_z\frac{\partial\phi}{\partial\eta} + \zeta_z\frac{\partial\phi}{\partial\zeta}$$

Poisson equation:

$$\xi_x\frac{\partial U'_x}{\partial\xi} + \eta_x\frac{\partial U'_x}{\partial\eta} + \zeta_x\frac{\partial U'_x}{\partial\zeta} + \xi_y\frac{\partial U'_y}{\partial\xi} + \eta_y\frac{\partial U'_y}{\partial\eta} + \zeta_y\frac{\partial U'_y}{\partial\zeta} + \xi_z\frac{\partial U'_z}{\partial\xi} + \eta_z\frac{\partial U'_z}{\partial\eta} + \zeta_z\frac{\partial U'_z}{\partial\zeta} = -1$$

(10)

where,

$$U'_x = \xi_x\frac{\partial\phi'}{\partial\xi} + \eta_x\frac{\partial\phi'}{\partial\eta} + \zeta_x\frac{\partial\phi'}{\partial\zeta}$$

$$U'_y = \xi_y\frac{\partial\phi'}{\partial\xi} + \eta_y\frac{\partial\phi'}{\partial\eta} + \zeta_y\frac{\partial\phi'}{\partial\zeta}$$

$$U'_z = \xi_z\frac{\partial\phi'}{\partial\xi} + \eta_z\frac{\partial\phi'}{\partial\eta} + \zeta_z\frac{\partial\phi'}{\partial\zeta}$$

3.2. Spatial discritization

3.2.1. Baseline solver

The baseline solver employs standard upwind schemes to estimate the advection analogous terms in the Eikonal and Hamilton-Jacobi equations; and 2nd order central difference scheme to estimate the Laplacian term in the H-J and Poisson equations. For example, the first term in the LHS of Eikonal equation (Eqn.8), $\hat{U}\phi_x$, is discretized using the up-wind scheme as follows:

$$\hat{U}\phi_x = 0.5(\hat{U} + |\hat{U}|)B + 0.5(\hat{U} - |\hat{U}|)F$$

(11)

where,

$$B = \frac{\phi_i - \phi_{i-1}}{\Delta\xi_{i-1}} \quad and \quad F = \frac{\phi_{i+1} - \phi_i}{\Delta\xi_{i+1}}$$

(12)

5

Here, 'i' corresponds to the grid point index in the ξ-direction. The terms $\frac{\partial \phi}{\partial \xi}$, $\frac{\partial \phi}{\partial \eta}$ and $\frac{\partial \phi}{\partial \zeta}$, which are required to compute $\hat{U}$, are estimated as follows:

$$\frac{\partial \phi}{\partial \xi} \approx n_{i-1}F + n_{i+1}B \tag{13}$$

$$n_{i-1} = 0.25(1 + \mathrm{SIGN}(1, F + B)(1 + \mathrm{SIGN}(1, B)) \tag{14}$$

$$n_{i+1} = 0.25(1 - \mathrm{SIGN}(1, F + B)(1 - \mathrm{SIGN}(1, F)) \tag{15}$$

Similar formulation is employed to estimate the metric terms and the derivatives of ϕ in other coordinate directions, $\frac{\partial \phi}{\partial \eta}$ and $\frac{\partial \phi}{\partial \zeta}$.

3.2.2. Enhanced solver

In the enhanced solver, high-order central difference schemes namely explicit second order (E2), explicit 4th order (E4), compact 4th order (C4) and compact 6th order (C6) schemes are used to discretize all the spatial derivatives and metric terms in the governing equations. Implementation of these schemes was straightforward as most of the subroutines were reused from the in-house high-fidelity fluid flow solver COMPSQUARE [17, 18, 19]. Following [7], the first-order derivative of a variable ϕ in the interior domain is computed as follows:

$$\alpha\phi'_{i-1} + \phi'_i + \alpha\phi'_{i+1} = b\frac{\phi_{i+2} - \phi_{i-2}}{4\Delta\xi} + a\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta\xi} \tag{16}$$

Here, ϕ' represents the derivative of ϕ in ξ direction. While the constant α is zero for explicit schemes (E2 and E4), it is non-zero for compact schemes (C4 and C6). Values of the scheme-specific coefficients 'a' and 'b' and the formulation of derivatives at the boundaries can be found in Ref.[7].

3.3. Solution convergence

For steady cases, a pseudo time derivative, $\frac{\partial \phi}{\partial t^*}$, has been added to the LHS of all the three differential equations (Eqn.17 18 & 19). An explicit 4-stage 4th order Runge-Kutta (RK) [7] is used to drive the solution to a steady state in pseudo-time, t^*. In addition, local-time stepping is used to accelerate the convergence.

$$\frac{\partial \phi}{\partial t^*} + \mathbf{U} \cdot \nabla\phi = 1 \tag{17}$$

$$\frac{\partial \phi}{\partial t^*} + \mathbf{U} \cdot \nabla\phi = 1 + (\nu\phi)\nabla^2\phi \tag{18}$$

$$\frac{\partial \phi}{\partial t^*} - 1 = \nabla^2\phi \tag{19}$$

Following convergence criterion is used to arrive at steady state solution:

$$\left|\frac{\phi - \phi_o}{L_{ref}}\right|_{max} \leq 10^{-5} \tag{20}$$

Here, ϕ and ϕ_o correspond to the wall distance values in the current and previous pseudo time iterations. The solution is converged until the maximum change in the wall distance value during a pseudo time step

6

is below 10^{-5} in the entire domain. For the unsteady cases, where the body(s) or the wall(s) move w.r.t time, the boundary conditions in the computational domain are updated at every physical time step and ϕ is repeatedly computed. Between two physical time steps, pseudo iterations with local time-stepping are used to drive the solution to a steady state.

3.4. Filtering scheme

High frequency dispersion errors arising due to the non-dissipative nature of the high-order central schemes (E2, E4, C4 and C6), and the non-linear nature of the governing equations, can lead to convergence issues. To counter this, an implicit 10th order spatial filter is used to stabilize the solution along all the directions after each pseudo time step. Following [7], the implicit filtering operation of a variable Φ along a direction i is given as:

$$\alpha_f \hat{\phi}_{i-1} + \hat{\phi}_i + \alpha_f \hat{\phi}_{i+1} = \sum_{n=0}^{n=N} \frac{a_n}{2} \left(\phi_{i+n} + \phi_{i-n} \right) \tag{21}$$

where ϕ and $\hat{\phi}$ are unfiltered and filtered functional values respectively and α_f is the filtering coefficient. $2N$ is the order of filter which uses a $2N+1$ point stencil. A 10th order filter ($N=5$) is used in the current work at interior points and α_f in the current work is varied between 0.48 and 0.495 for different test cases. The values of filter coefficients, a_n, at the interior and boundary points are taken from Visbal et al. [7]. Following their strategy, the filtering order is gradually reduced up to second order at the boundaries.

3.5. Boundary Conditions and Initialization

Dirichlet boundary condition is imposed for the grid points on the wall, where the value of the wall distance is set to zero. Wall distance ϕ, is also set to zero at all the grid points lying within the volume of the solid body/bodies.

$$\phi_{wall} = 0 \tag{22}$$

Neumann boundary condition is imposed at the far-field boundaries, where the gradient of wall distance is set to zero.

$$\frac{\partial \phi}{\partial n} = 0 \tag{23}$$

Here, n is the direction normal to the boundary. The wall distance values are initialized to zero within the domain before starting the simulation. Figure 1 shows the flow charts with the details of the algorithm for the baseline and enhanced solver.

4. Test cases

The implementation of both the baseline and enhanced solvers has been tested on several steady and unsteady test cases listed in Table-1. Cases 1 and 2 are used to validate the baseline solver of which Case-2, chosen from [4], demonstrates the capability of the solver in estimating the wall-distance field around the complex bodies within the computational domain. Cases 3, 4 and 5 are used to validate the enhanced solver and compare the results against the baseline solver. Case-6 is solved to examine the effect of curvature on the accuracy of wall distances computed using H-J approach. Results using standard H-J and standard H-J based on the curvature corrections introduced by Nakanashi [22] are compared. We also discuss the modifications to the Nakanashi's [22] approach, which have improved the accuracy further. Cases 7 and 8 respectively simulate a piston-cylinder arrangement and a cube bouncing between parallel walls. These test cases demonstrate the efficacy of the unsteady solver where the wall-distances between the surfaces vary with time. In addition, we have also simulated the wall distance evolution inside a solid rocket motor with a progressively burning propellant grain.
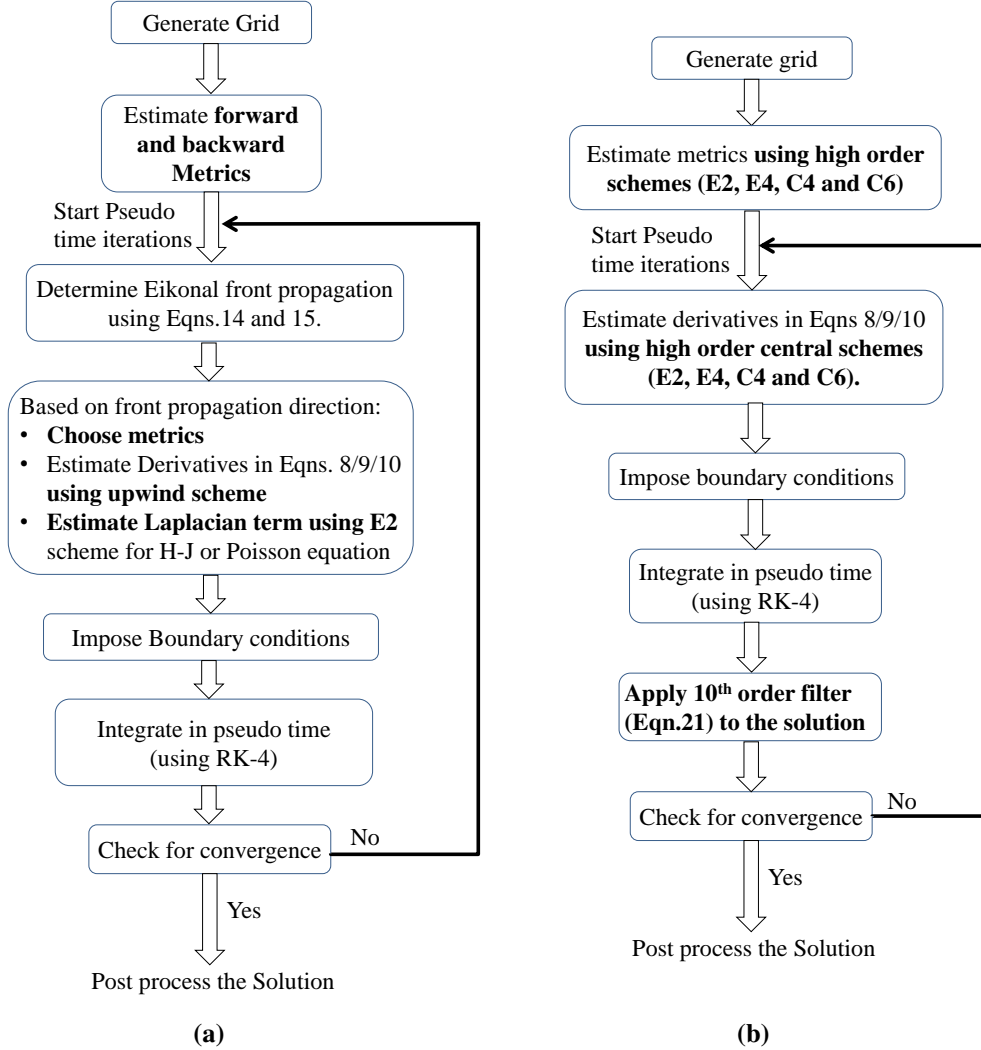
7

Figure 1: **Algorithm flow chart for (a) Baseline UW solver and (b) Enhanced high-order solver**

5. Results

In this section the baseline solver which employs first order up-wind scheme is validated on cases 1 and 2 (3-D box and Complex geometries). Subsequently, the enhanced solver is validated on rest of the test cases. For all the test cases, we compare the wall-distance and the error field computed from enhanced solver against the exact results and those obtained using the baseline solver. The exact solution was found analytically based on the case geometry, as most of the test cases used in this work are canonical in nature. Computational time (CPU time) required by each of these methods are also compared.

5.1. Validation of the baseline solver

Figures 2(a,b,c) depict the wall-distance field computed within a 3D box (case-1) on Z = 5 plane. Corresponding errors w.r.t the exact wall-distance are shown in Figures 2(d,e,f). We compare the solutions predicted using the Eikonal, H-J and Poisson based approaches. For clarity, Figure 2(g) also shows the wall-distance on a vertical line marked in Figure 2(a). The errors are negligible close to the walls for all

Case No.	Case name	Geometry	Grid	Boundary conditions & Additional details
1	3-D box	10×10×10 3-D cube	41×41×41	All faces are walls
2	Complex geometry	10×10 2-D square domain with multiple geometric shapes.	101×101	All shapes treated as walls
3	Flat plate	10×10 2-D square domain	32^2, 64^2, 128^2, 256^2	Bottom edge treated as wall
4	Parallel plates	10×10 2-D square domain	32^2, 64^2, 128^2, 256^2	Top and bottom edges treated as walls.
5	2-D box	10×10 2-D square domain	32^2, 64^2, 128^2, 256^2	All edges treated as walls
6	Circular bump	2.4×3.6 domain with a circular bump.	41×41	Bump on bottom surface treated as wall
7	Piston cylinder arrangement	2-D piston executing vertical harmonic motion inside 5×10 cylinder	41×131	Oscillating piston Frequency: 1.5π Hz Amplitude: 3 units
8	Bouncing cube	2-D cube of size 1×1 bouncing inside a 10 x 10 domain	64×64	Top and bottom surfaces treated as walls Cube looses 20% of its total momentum after each bounce

Table 1: **Case set-up, boundary conditions and grid details for various canonical test cases.**

the approaches. However, it is evident that the highest errors are recorded at the centre of the box and across the diagonals, particularly with the Poisson equation. On the other hand, the predictions of both the Eikonal and H-J are much more accurate, with the Eikonal solution being the most accurate throughout the domain. As noted in section 2.3, the loss of accuracy away from the walls with H-J is attributed to the Laplacian term using the standard Γ formulation. Figure 3 also compares the computed wall distance contours for a test case with complex geometries embedded in a square domain (case-2). The wall-distance field obtained is in agreement with the results reported in [4] further validating the framework. It also demonstrates the robustness and performance of the current solver in handling geometries with complex shapes.

5.2. Effect of high order schemes and LAD formulation on the solution of Hamilton Jacobi Equation

Having validated the baseline solver using the upwind schemes, we have solved the same set of differential equations with the high-order central schemes inbuilt in the in-house solver COMPSQUARE. As demonstrated in the flow chart of Figure 1(b), the wall-distance solver has been enhanced to exploit the high order explicit/compact discretization schemes (E2,E4,C4,C6) with a tenth order filter (F10). The efficacy of the new LAD based formulation for Γ (Eqn. 7) in the H-J equation has also been examined. The performance of the enhanced solver, in terms of accuracy and speed, is reported in this section.

Figure 4(a) present the contours of wall distance (ϕ/L_{ref}) and the corresponding errors for the canonical test cases of 'Flat plate', 'channel' and '2-D box' respectively. For clarity, we have also included line plots of the wall distance (Figure 4(b)) and the errors (Figure 4(c)) along the vertical line (shown with dotted lines in the corresponding figures). From the error contours and line plots, it is evident that the predictions of the Pseudo HJ approach with LAD based formulation are in close agreement with the exact solution
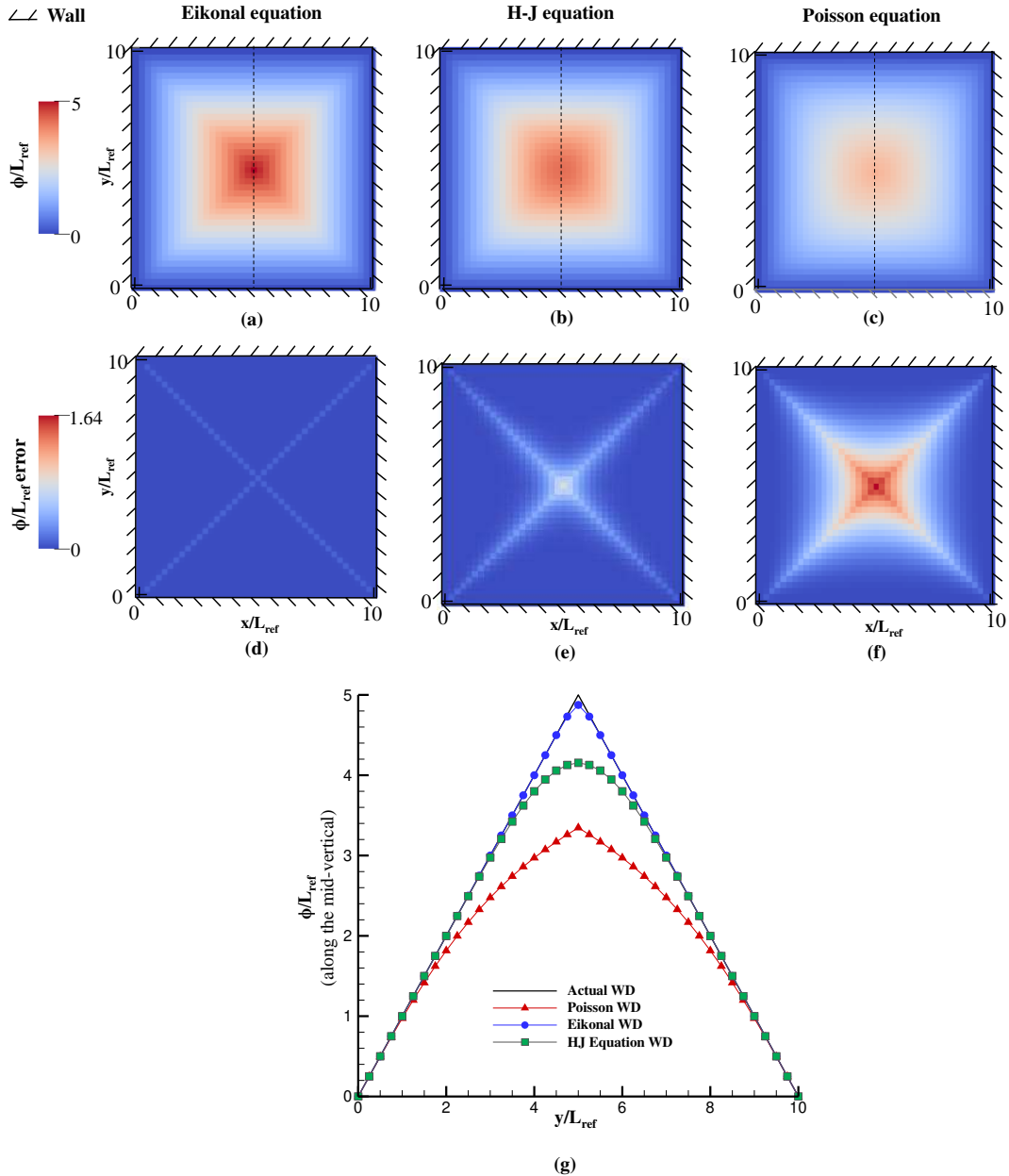
Figure 2: **Wall distance field (a,b,c) and the corresponding error field (d,e,f) for 3D box case (case-1) obtained by solving Eikonal, Hamilton-Jacobi equation and Poisson equation with the baseline solver (g) Comparison of wall distance along the mid-vertical line with the exact solution.**

for all the three test cases than the standard HJ. Selective amount of dissipation added using the LAD formulation (Eqn.5 and 7) has remarkably improved the accuracy while maintaining the stability. On the other hand, a marginal improvement in the accuracy is observed with high order schemes (E2, E4, C4 and C6) when compared to the UW scheme. To quantify the improvements in accuracy, simulations are carried out at different grid resolutions for the channel case. Subsequently, the global L^2 norm is estimated as

10

(a) Eikonal equation (UW) **(b)** H-J equation (UW) ($\varepsilon = 0.3$) **(c)** Poisson equation (E2)
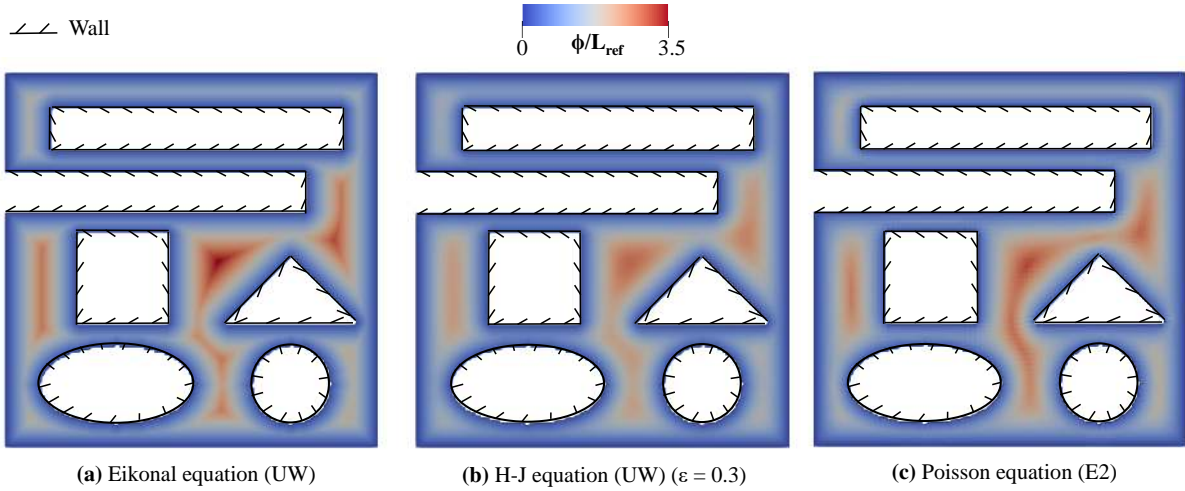
Figure 3: **Wall distance field obtained using (a) Eikonal, (b) Hamilton-Jacobi and (c) Poisson equations for complex geometries case (case-2).**

follows:

$$L^2 = \left(\Delta x \Delta y \sum_i \sum_j |\phi - \phi_{exact}|^2 \right)^{1/2} \tag{24}$$

Figure 5(a) compares the L^2-norm for the channel case at different grid resolutions. As expected, the benefit of high order schemes is much more noticeable on a coarser grid than on finer grids. In addition, when compared to the standard HJ formulation, the L^2-norm with LAD based formulation is almost an order of magnitude lower.

Figure 5(b) compares the computational time per 100 iterations for the baseline and enhanced H-J solvers. All the simulations are carried out on a single processor (Dell Optiplex 3070 tower using 3 GHz Intel Core i7-9700 CPU with 32 GB/2666 MHz DDR4 memory). Unlike central difference schemes, upwind schemes involve the repeated computation of the wave front direction at every iteration. Hence, the explicit schemes (E2 and E4) employing the standard and LAD based H-J formulation, are $\approx 2.8\times$ and $\approx 1.5\times$ faster than the baseline solver respectively. LAD based H-J formulation involves computing Γ during each iteration and are hence slower than the standard H-J formulation using explicit schemes. Interestingly, the speed of the baseline solver using UW schemes and the enhanced solver using compact schemes for the LAD based H-J equation are almost similar. To summarize, the advantage of using the high-order central difference schemes with filtering over the UW schemes, both in terms of the speed up and accuracy, can be appreciated while solving the H-J equation.
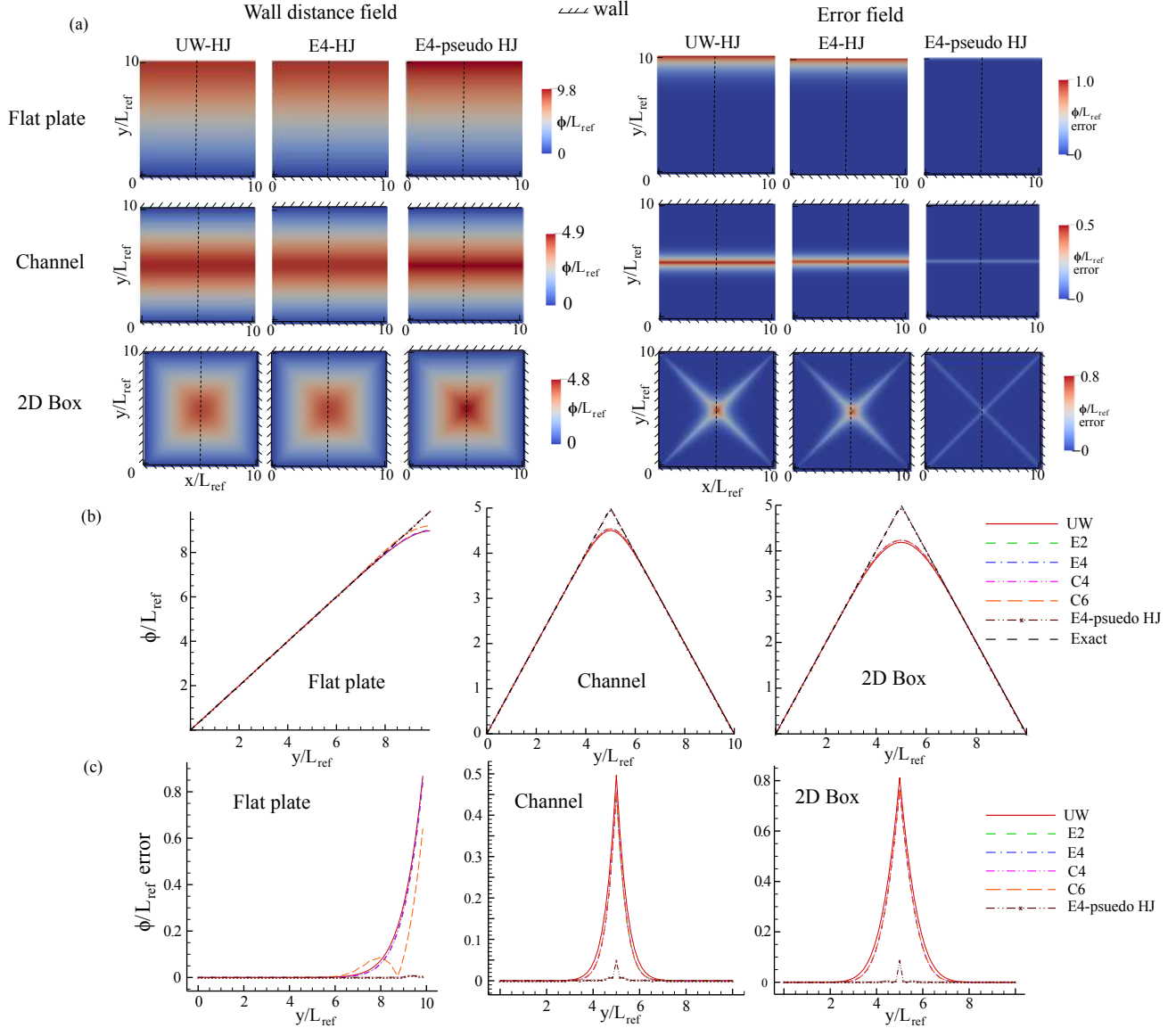
11

Figure 4: **(a) Contours of wall distance field and absolute wall distance error. Variation of (b) wall distance and (c) absolute wall distance error along mid vertical line (marked with dashed lines in contour plots) for flat plate, channel and 2D box cases.**

5.3. Effect of High order schemes on the solution of Eikonal equation

Inline with the analysis carried out for the H-J equation, we have tested the efficacy of high-order schemes to solve the Eikonal equation. Performance of various schemes is tested on the 'channel' case. Figures 6(a,b) show the wall distance field and the error contours in addition to the line plots along the domain center. It is evident that the wall distance predictions with UW and central difference schemes are in close agreement; and any improvement in accuracy due to high order is marginal. In fact, when compared to the H-J equation, the Eikonal equation is hyperbolic and high-order schemes are susceptible to stability issues due to the lack of the Laplacian term. Dispersion errors are hence evident at the centre of the channel (although away from the walls) where the fronts propagating from both the walls meet. Similar
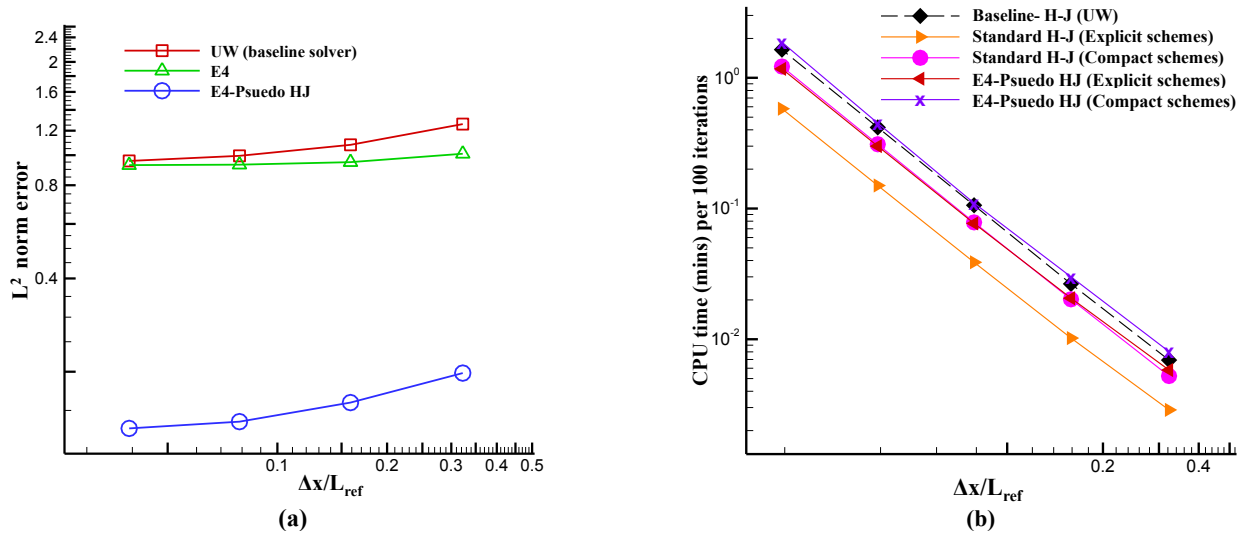
12

Figure 5: **(a)** L^2 **norm error with varying grid resolutions and (b) CPU time taken per 100 iterations with different schemes for channel case.**
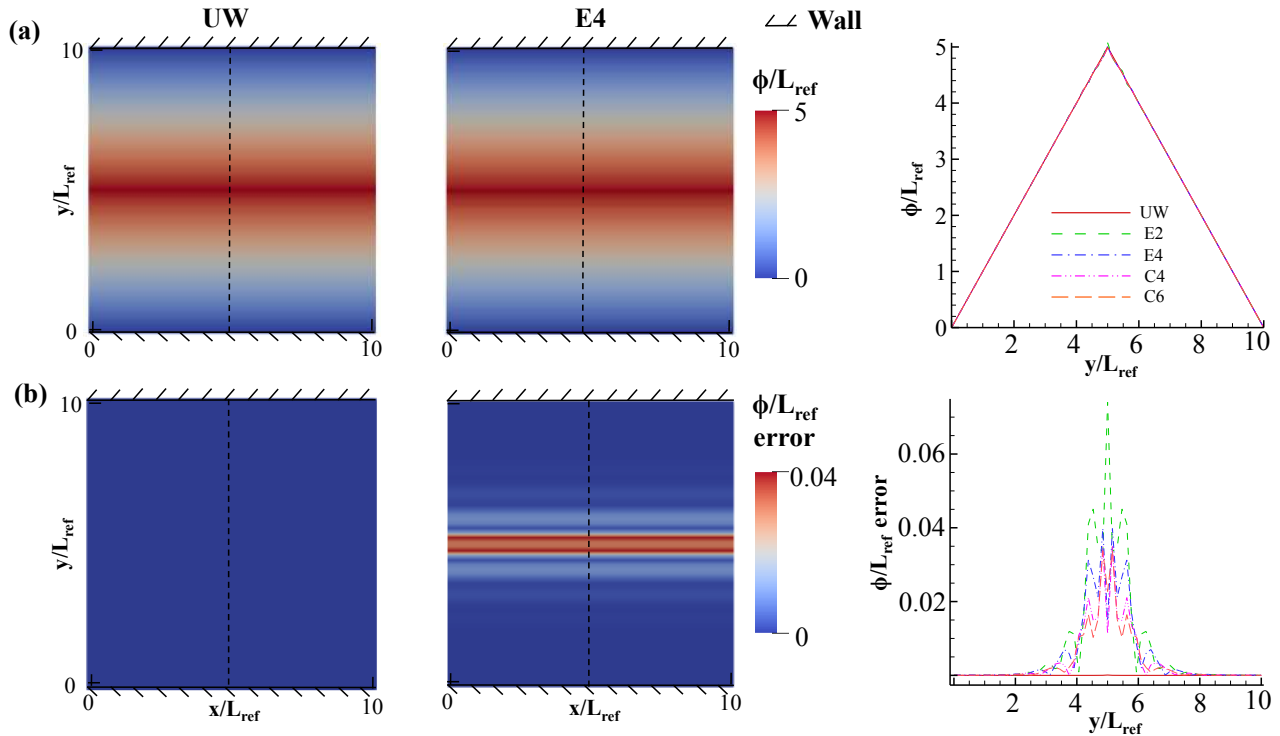

Figure 6: **Contours of (a) Wall distance field and (b) absolute wall distance error field for the channel case predicted by solving Eikonal equation with UW and E4 schemes. Corresponding line plots along mid-vertical line are also shown**

conclusion can be drawn from the bar charts shown in Figure 7(a), where we compare the L^2 norm error

13

of the baseline (UW) and enhanced solvers for three different test cases. The errors for the flat plate and channel cases are lower using the UW scheme when compared to the high-order schemes. For the 2D box case, the errors corresponding to UW and central-schemes are comparable and are predominant across the diagonals (similar to the errors shown in Figure 4). The results suggest that although the dispersion due to Gibbs phenomenon can be partially suppressed with filtering, the errors due to central schemes while solving the Eikonal equation still persist in the regions of steep gradients where the wave fronts meet. A clear advantage of central schemes is notable in Figure 7(b), where the computational time required for 100 pseudo iterations are compared for different schemes for the '2-D box' case. Due to the lack of the Laplacian term, computational cost associated with solving the Eikonal equation is relatively lower than H-J. The trends are similar to those observed in Fig.5(b) with the compact schemes (C4/C6) and Explicit schemes (E2/E4) being $\approx$ 2.3 to 3.5 times faster than the UW solver respectively.
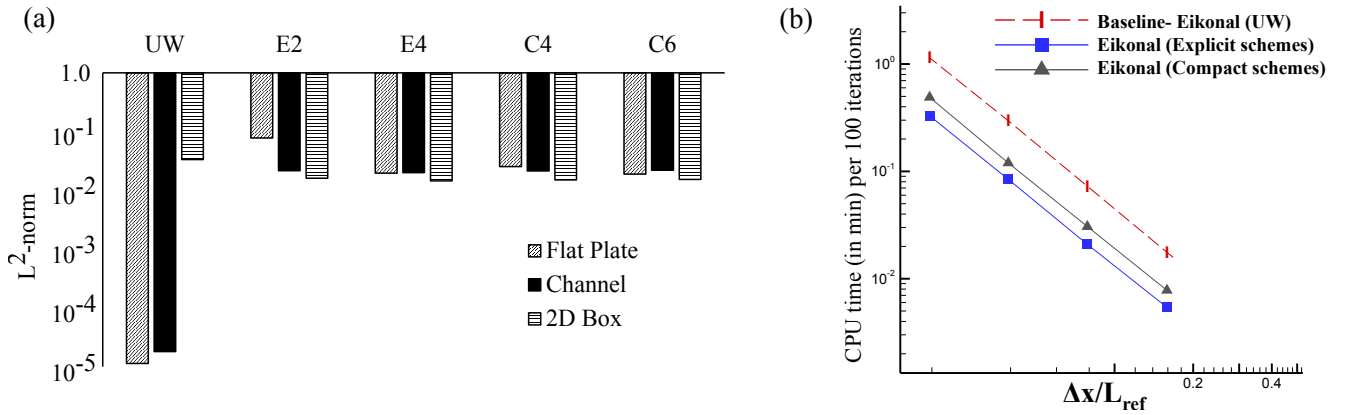

Figure 7: **Comparison of (a) L2 norm for flat plate, channel and 2D box cases using different schemes (b) CPU time taken per 100 iterations by the baseline and enhanced solver for 2D box case.**

To further demonstrate the speed up using central-schemes, Figures 9(a,c) show the convergence of the maximum residual given by Eq. 19 with CPU time for the channel case. In addition, Figures 9(b,d) compare the convergence of L2 norm error (see Eq. 24) with time for HJ and Eikonal schemes. From the plots, it is clear that the computational time associated with the upwind schemes is generally higher when compared to the central schemes. In particular, the explicit schemes offer more computational benefit and are almost 2× faster than the compact/upwind schemes to converge to the same level of accuracy. This is attributed to the repeated computation of the wave front direction in upwind schemes. The effect on accuracy with different schemes is also minimal while solving the HJ equations. UW schemes are certainly suitable for the Eikonal based approach in terms of accuracy, as they can converge to lower values of L2 norm error (see Figure 9(d)). However, as noted earlier, the compact (C4/C6) and Explicit schemes (E2/E4) are $\approx$ 2.3 to 3.5 times faster than the UW schemes respectively.

Following Tucker [1], the error histograms for the HJ, Pseudo-HJ and Eikonal equations are estimated. The percentage errors are estimated as $(d_{act} - d)/d_{act} \times 100\%$ where d and d_{act} are the predicted and true wall-distances. Figure 9 compares the histogram for different schemes. Negative errors represent over-prediction of wall distance w.r.t the true value and positive errors represent under-prediction. With HJ based approach, the scatter in the error is within $0 - 10\%$ with a positive bias (Fig. 9(a,b)). This bias is attributed to the under-prediction of wall-distance away from the walls (see Fig. 4). Figure 9(c) shows a significant reduction in the scatter of the error using Pseudo-HJ. On the other hand, the error with Eikonal is within $\pm 0.2\%$ and is symmetrical than the HJ based approach (Fig. 9 (d,e,f)). Although the relative
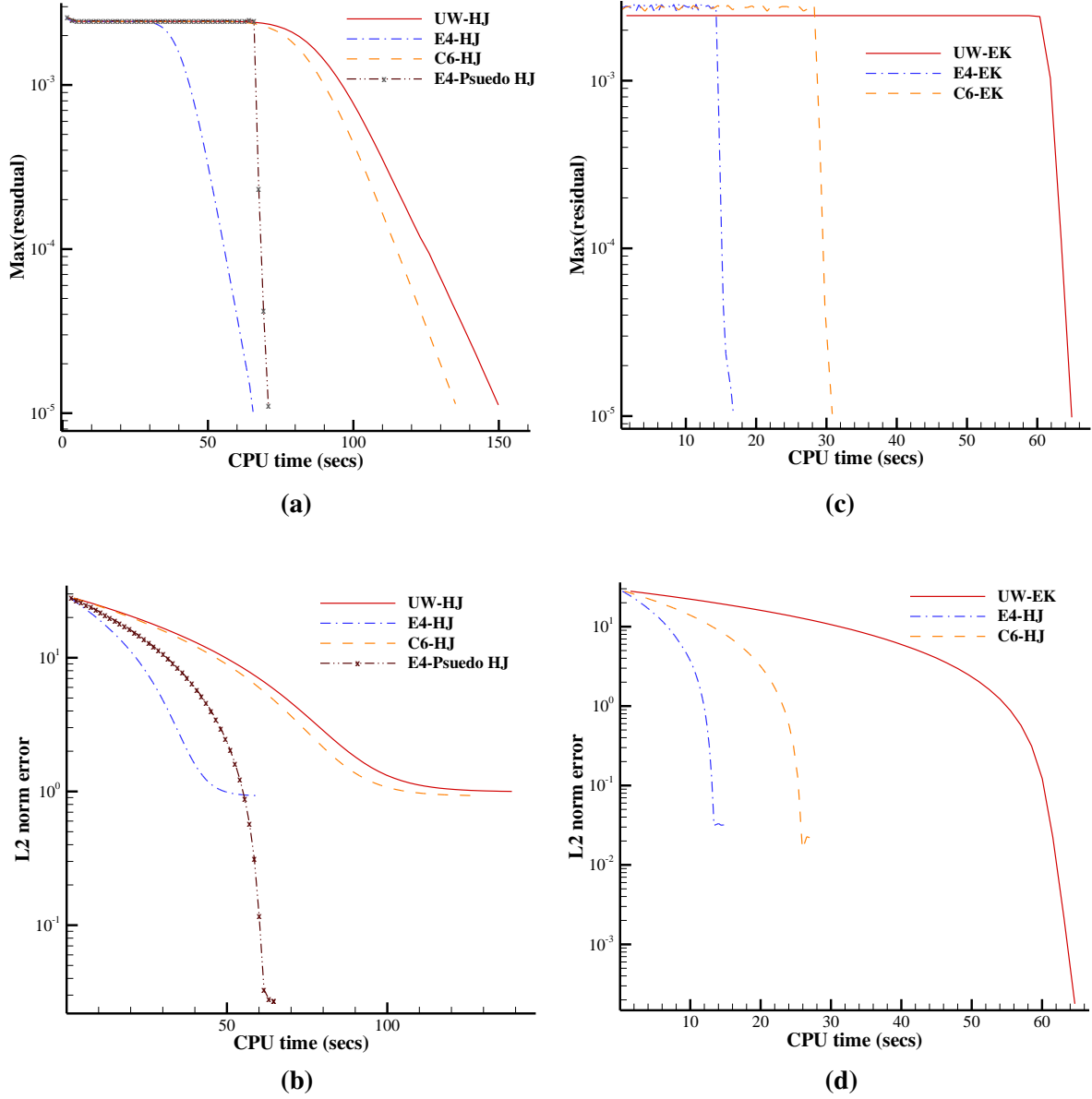
Figure 8: **Convergence histories of maximum residual (a,c) and L^2 norm error (b,d) for channel case using upwind and high-order schemes for Hamilton Jacobi (a,b) and Eikonal (c,d) equations.**

error magnitude is lower for all the schemes, the scatter in the error is much lower for UW-EK compared to the central-schemes.

5.4. Effect of wall curvature and Nakanishi's modification

The Laplacian term in H-J equation is dependent on the wall curvature and can assume very high or low values near convex and concave regions of the domain respectively. As a result, the H-J equation largely deviates from the Eikonal equation which results in high wall distance errors. Mathematically, $|\nabla^2\phi| \gg 0$ for convex regions. Hence, the RHS of the H-J equation $|\nabla\phi| = \sqrt{1 + \Gamma\nabla^2\phi}$ will be much more greater than 1 in highly curved regions, there by significantly deviating from the Eikonal equation. For
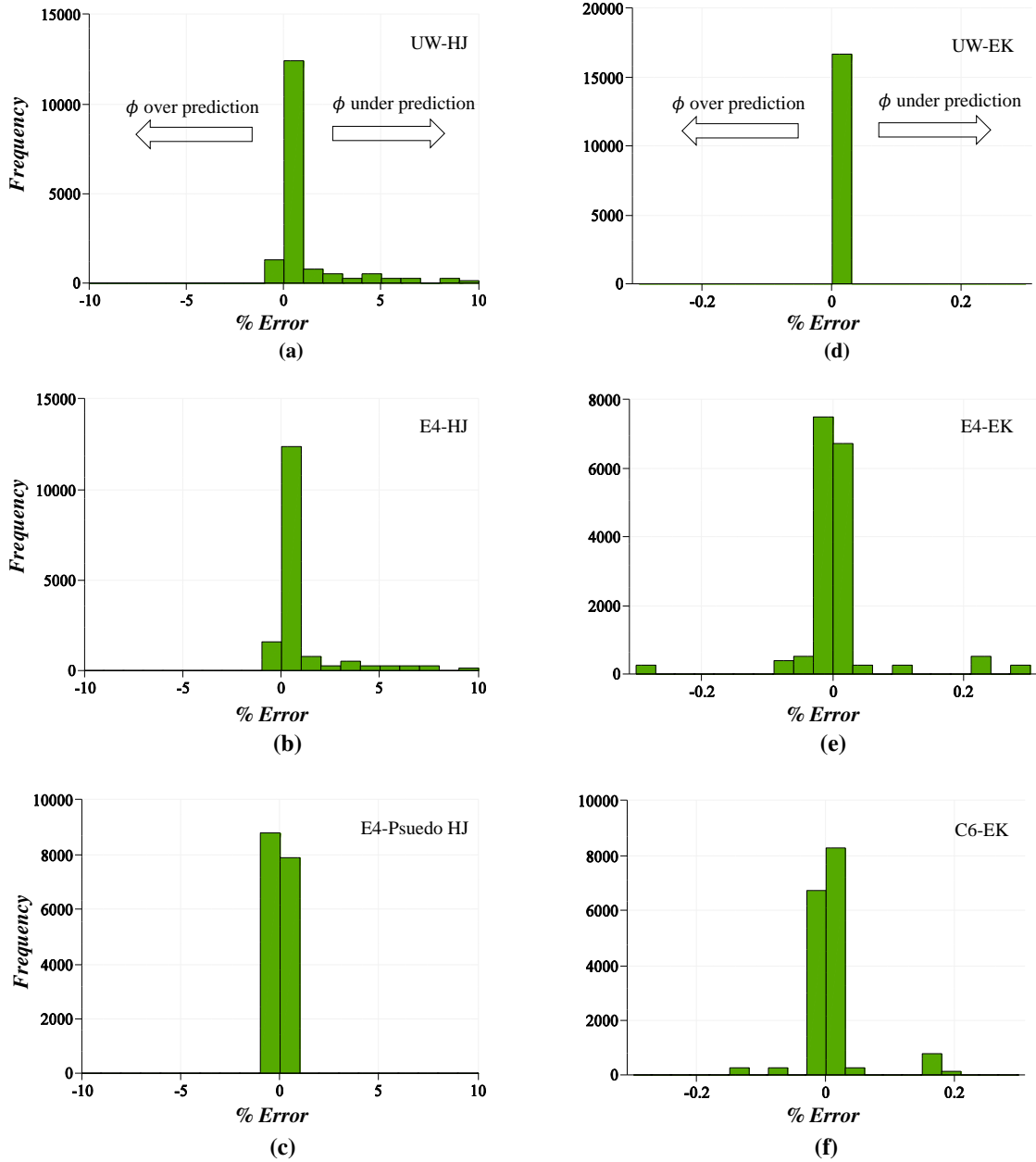
Figure 9: **Histograms of percentage error in wall distance estimates for channel case corresponding to Hamilton-Jacobi and Eikonal equations computed using upwind and high order schemes. The frequency on y-axis refers to the total number of solution points in the entire computational domain that constitute the percentage error range represented on the x-axis.**

example, this effect can be seen at the airfoil leading and trailing edges. To counter this, Nakanishi [22] has proposed modifications to the original H-J equation. A curvature (κ) based correction is introduced into the formulation which can eliminate the influence of convex wall curvature on wall distance solution.

The modified form reads as follows.

$$\mathbf{U} \cdot \nabla\phi = 1 + (\epsilon\phi + v)\left\{\nabla^2\phi - MAX(0, |\nabla\phi|\kappa)\right\} \tag{25}$$

The expressions for ν (which is introduced to enhance stability) and κ are as follows:

$$v = 0.001(1 - |\nabla\phi|)^2$$

$$\kappa = \nabla \cdot \boldsymbol{n}, \quad \boldsymbol{n} = \frac{\nabla\phi}{|\nabla\phi| + \varepsilon_0}$$

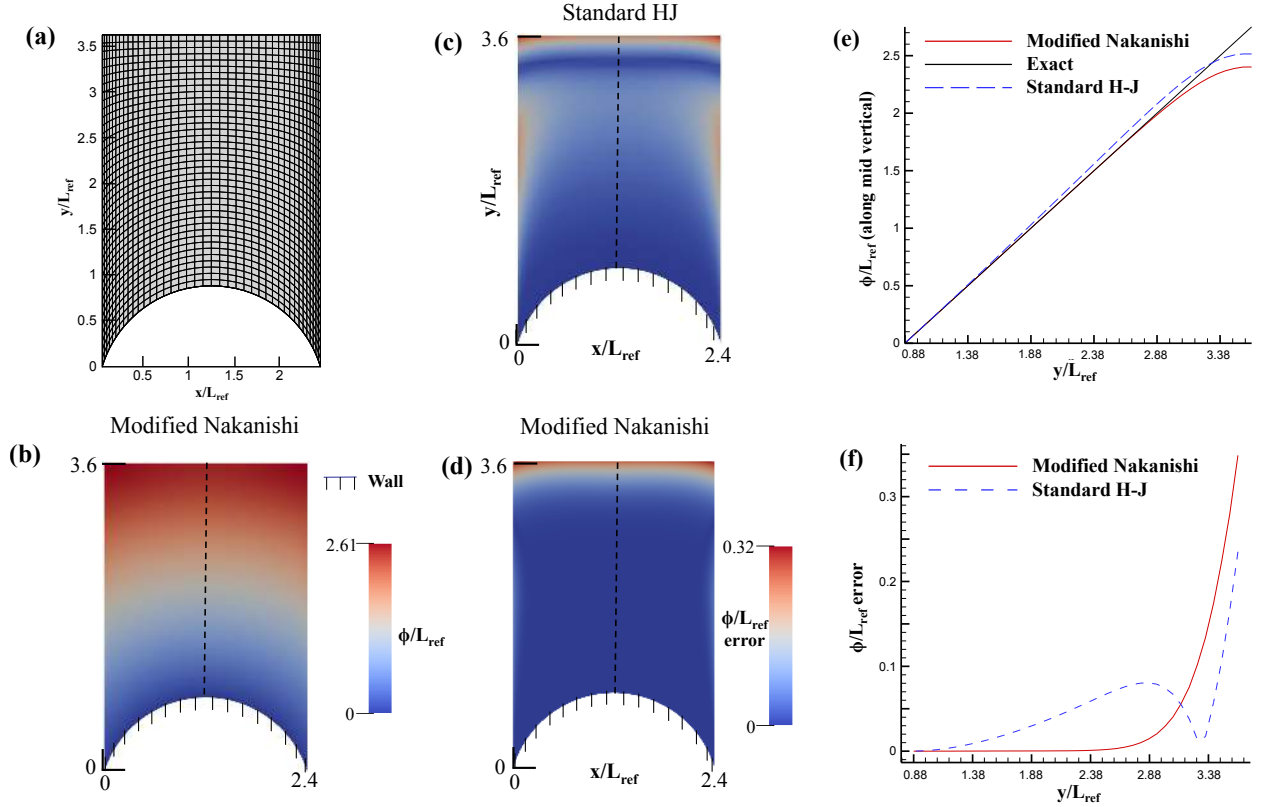Figure 10: **(a) Computational grid used for the circular bump case; (b) Wall distance contours (ϕ/L_{ref}); Contours of wall distance error ($\phi/L_{ref}error$) using (c) standard H-J and (d) Modified Nakanishi approach; Comparison of (e) wall distance and (b) wall distance error obtained along the mid vertical of the computational domain. Simulations use E4 scheme with $\alpha_f = 0.495$.**

In the absence of filtering, the simulations with this formulation using E2 and E4 schemes are observed to be unstable due to large dispersion errors. To counter this, the value of ϵ can be increased, say to 1, as proposed by Nakanishi [22]. However, higher values of ϵ (a) reduce the wall distance accuracy away from the walls and (b) reduce the stable pseudo time-step (Δt^*) thereby increasing the simulation time. To account for these accuracy and convergence issues, we have reduced the value of ϵ to 0.2 and used filtering (with $\alpha_f = 0.495$). We refer this approach as 'Modified Nakanishi' in the plots.

Figure 10 shows the computational domain for the 'circular bump' test case along with the contours of wall distance and absolute errors obtained using the standard H-J and modified Nakanishi approaches. From the wall distance and error plots shown in figures 10 (e) and (f), the effect of Nakanishi's formulation

17

in mitigating the wall distance error can be clearly appreciated. The wall distance deviates from the exact solution using standard H-J approach even closer to the walls. On the other hand, the modified Nakanishi solution rectifies these curvature related issues and predicts accurate wall distances in most of the domain, particularly near the walls. It should be noted that Nakanishi's approach is recommendable for the applications where accurate wall distances are required at all the locations in the domain (CAD/automated meshing/computational interface for overset grids/acoustic ray tracing,etc). However, in the context of turbulence modelling, the Laplacian operator in the original formulation by Tucker et al. [2] has been deliberately tuned to exaggerate the wall-distance, $\phi > d_{true}$, near the fine convex features (thin wires) or convex corners to remedy the excessive turbulence dissipation. On the other hand, through $\phi < d_{true}$ at a concave corner, the Laplacian term naturally accounts for the damping effects due to extra walls. Similar benefit is also observed while using the Poisson-equation based approach for computing wall-distances for turbulence modelling [1].

5.5. Unsteady cases

In this section, we discuss the results from the unsteady test cases like 'piston cylinder arrangement', 'bouncing cube' and 'propellant grain burnback in solid rocket motor' using the Eikonal and H-J approach with E4 scheme are presented here. The computational set-up, grid and boundary conditions for these cases are described in Section 4.

Canonical Cases: Figures 11(a) and 12(a) show the temporal variation of the wall-distance field for the 'piston-cylinder arrangement' and 'bouncing cube' respectively. The conventional method of computing wall distances using search methods are particularly expensive in these kind of scenarios where the walls are moving. Soon after the first physical time-step, we have observed a drastic reduction in the simulation time for the subsequent time steps. This is attributed to the fact that the solution at the previous physical time step provides a close enough initial wall-distance estimate for the new physical time step thereby accelerating convergence. Figures 11(b) and 12(b) compare the temporal variation of wall-distance against the exact solution recorded at probes marked in Figures 11(a) and 12(a). The predictions of both Eikonal and H-J solutions are in good agreement with the analytical value. For the bouncing cube case, the cube looses its energy by 20% every time it bounces off the ground, until it settles on the bottom surface. Hence, in Figure 12(b), an oscillatory convergence of the wall-distance to a steady value with increasing time is observable. Under the steady state, the probe is at a location which is farthest from all the walls. It has been demonstrated in Fig. 12 that standard H-J equation is less accurate away from the walls. At t/τ = 10-15, a notable error between the analytical and predicted wall distance persists due to the Laplacian operator in the standard H-J equation. It is also apparent from Figures 11(b) and 12(b) that the results using the new H-J formulation with LAD are much more accurate than the standard H-J and are in encouraging agreement with the Eikonal and exact solutions.

Burnback of Propellant Grain: One of the critical applications of unsteady wall-distances is in Solid propellant Rocket Motors (SRM), where the thrust-time or pressure-time characteristics of SRM has to be determined as a part of the design process. In SRM, a propellant grain of a given cross-sectional shape (star/tubular/double anchor/dendrite,etc) is burnt in the combustion chamber. For example, Fig. 13 shows a 8 legged star shaped propellant grain burning radially outward. The equilibrium pressure generated inside the combustion chamber, P_c, during the burning process can be obtained by equating the rate at which hot gases are generated from the propellant surface (of instantaneous surface area S_b) to the rate at which hot gases leave the nozzle downstream (of throat area A^*). It is approximately given as:

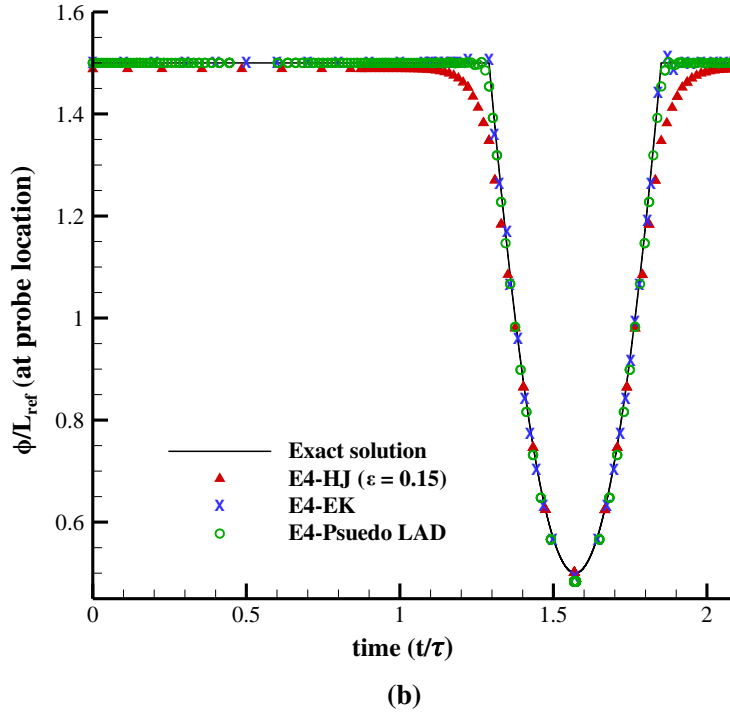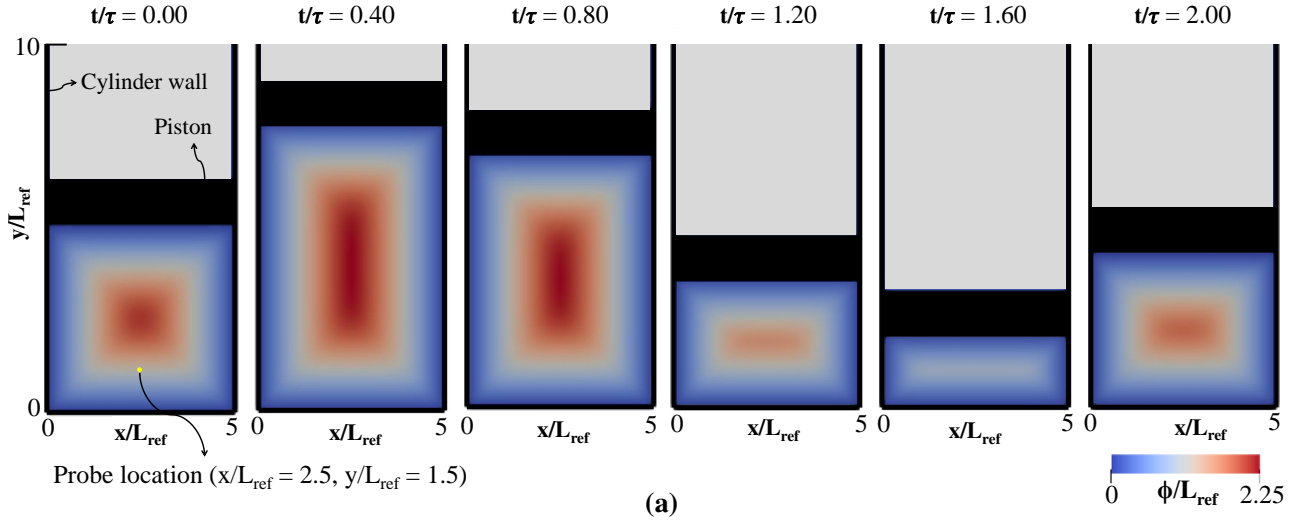$$P_c = \left(aC^*(\rho_p - \rho_0)\frac{S_b}{A^*} \right)^{\frac{1}{1-n}} \tag{26}$$

Figure 11: **(a) Temporal evolution of wall distance field inside the piston-cylinder arrangement (case-7) computed using H-J equation (with epsilon = 0.15) (b) Comparison of wall-distance evolution at the probe location shown in the first snapshot with the exact solution.**

Here, C^* is the characteristic velocity which is primarily a function of combustion chamber properties, ρ_p is the propellant density and ρ_0 is the instantaneous density of the hot gases. a and n are empirical constants (see [24] for further details). It is evident from Eq. 26 that the instantaneous pressure P_c within the SRM is a direct function of the instantaneous burning surface area of the propellant grain, S_b. Determining the temporal change in the shape of the propellant grain (and hence S_b) is crucial to accurately estimate the corresponding pressure-time characteristic. A level-set equation given below can
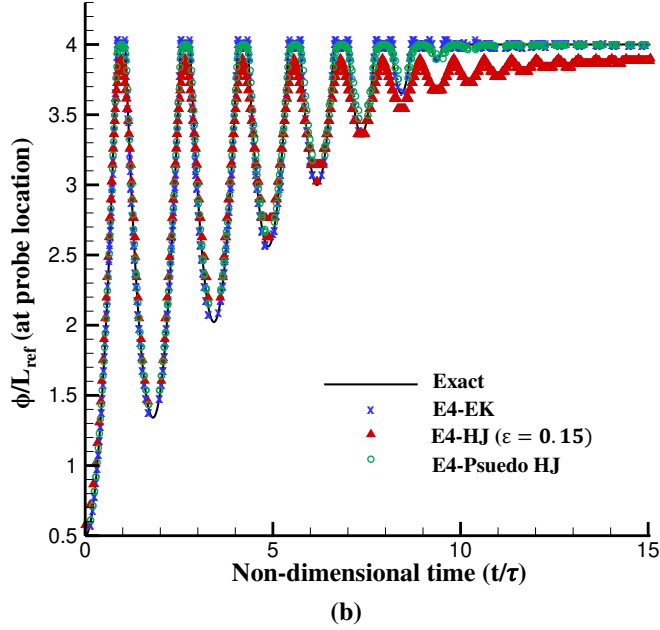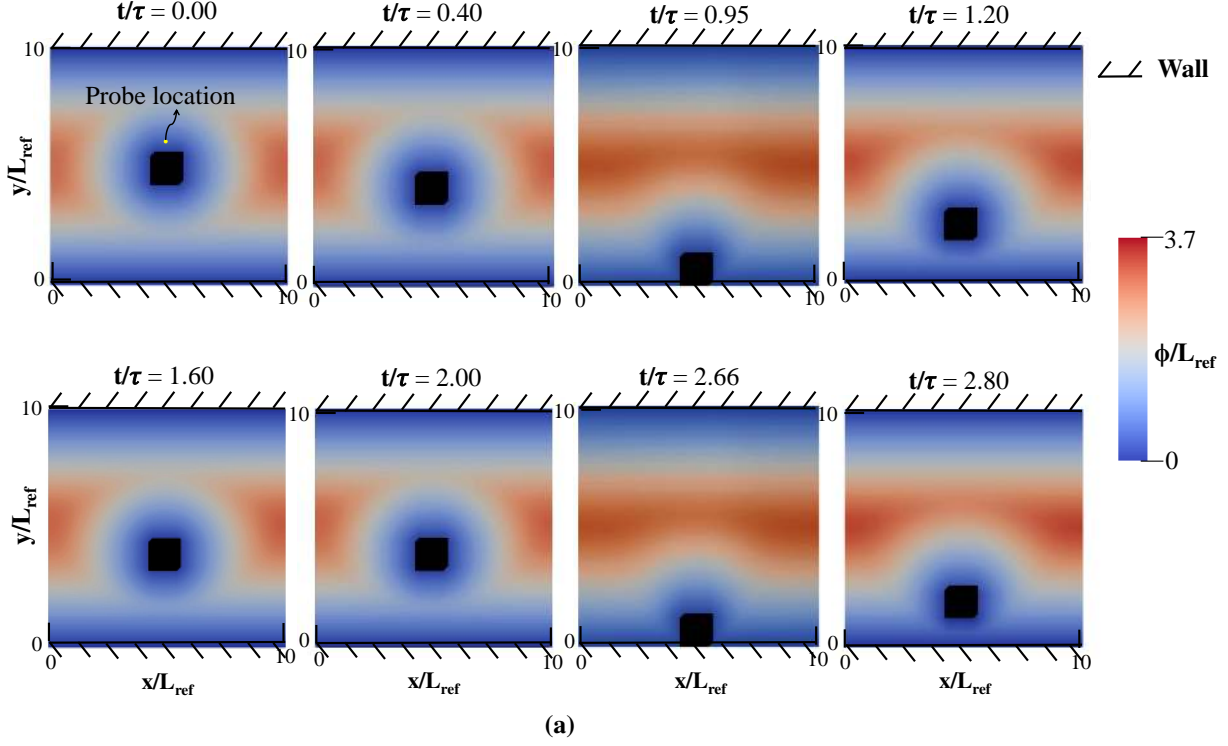
Figure 12: **(a) Temporal evolution of wall distance field around the bouncing cube (case-8) computed using H-J equation (with epsilon = 0.15) (b) Comparison of wall-distance evolution with the exact solution at the probe location shown in the first snapshot.**

be solved to determine the instantaneous shape of the grain.

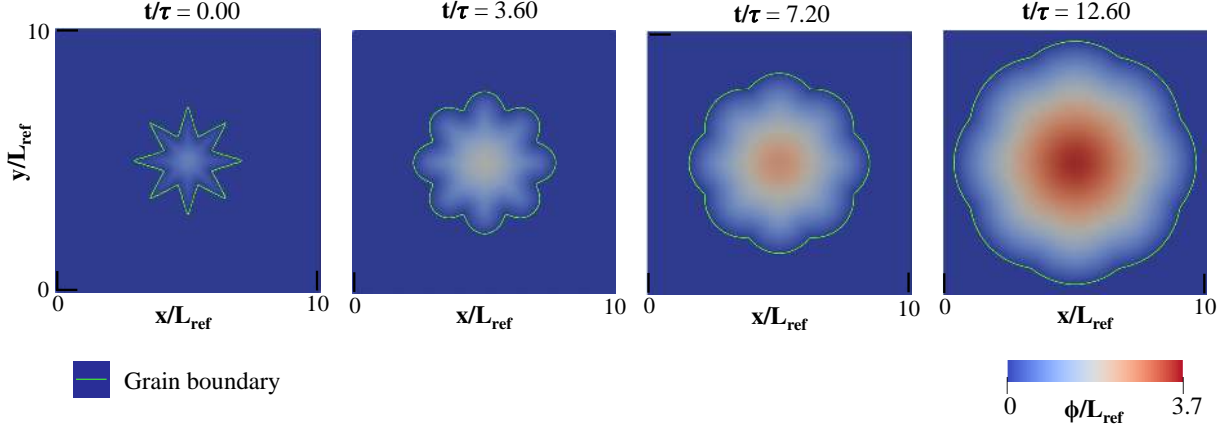$$\frac{\partial \phi_s}{\partial t} + F|\nabla \phi_s| = 1 \tag{27}$$

Figure 13: **Temporal evolution of wall distance field inside the combustion chamber of a solid rocket motor (with a 2-D eight legged star like propellant grain in it) computed using H-J equation.**

Here, ϕ_s is the signed wall-distance and F is the regression speed of the grain boundary along the local normal. A zero level-set of the signed wall distance (iso-surface of $\phi_s=0$) represents the grain interface at that specific time instant. The gradient of signed wall distance is set to zero at all the other domain boundaries. Without loss of generality, we demonstrate the evolution of a grain boundary for a constant regression speed of $F = 0.3$. The effect of varying regression speed, as discussed in Wei et al. [25], can also be accommodated in the current framework. The solution to this level-set equation is obtained in similar lines to that discussed in Sections 3.1 and 3.2. Firstly, $\phi_{s,\xi}$, $\phi_{s,\eta}$ and $\phi_{s,\zeta}$ are estimated using the upwind schemes described in Eqns. 13-15. Subsequently, $\phi_{s,x}$, $\phi_{s,y}$ and $\phi_{s,z}$ are estimated using the corresponding metric terms. Finally, $|\nabla\phi_s|$ in Eq. 27 is determined as follows:

$$|\nabla\phi_s| = (\phi_{s,x}^2 + \phi_{s,y}^2 + \phi_{s,z}^2)^{0.5} \tag{28}$$

The final discretized equation is integrated in time using the standard explicit RK4 time-integration scheme. Figure 13 shows the temporal evolution of a dendrite grain configuration. Figure 14 shows the unsteady burning of a dendrite grain at different time instants. Contours show the signed wall-distance field and the lines show the grain boundary corresponding to a zero level set. The burning is radially outward and progressive. The instantaneous burning surface area of the propellant grain, S_b, increases with time and the sharp corners gradually smooth out. Beyond $t/\tau \approx 0.96$, S_b decreases during the web burnout phase.

To initialize the aforementioned level-set procedure at $t/\tau = 0$, a signed wall-distance field, ϕ_s, must be computed for the initial dendrite grain configuration using any of the approaches (Poisson/HJ/HJ LAD/Eikonal) discussed in the previous sections. Subsequently, the level-set equation (Eq. 27) can be solved to track the evolution of the grain boundary. Figure 15 compares the effect of different initialization procedures on the evolution of the level-set contours. For comparison, a wall-distance field initialized by solving the Eikonal equation using upwind schemes is overlaid. Both the Eikonal and HJ-LAD based initialization, obtained using explicit 4th order schemes, show an encouraging agreement against the reference. On the other hand, discrepancies are evident in the level-set contours when the wall-distance field is initialized either using the baseline HJ or Poisson equations due to their inaccurate predictions away from the wall. Figure 16 compares the corresponding contours of the absolute error for different initialization strategies. Clearly, the L_1 and L_∞ errors using Eikonal and LAD based HJ schemes are an order of magnitude lower than the level-set contours predicted using Poisson/baseline HJ based initialization.
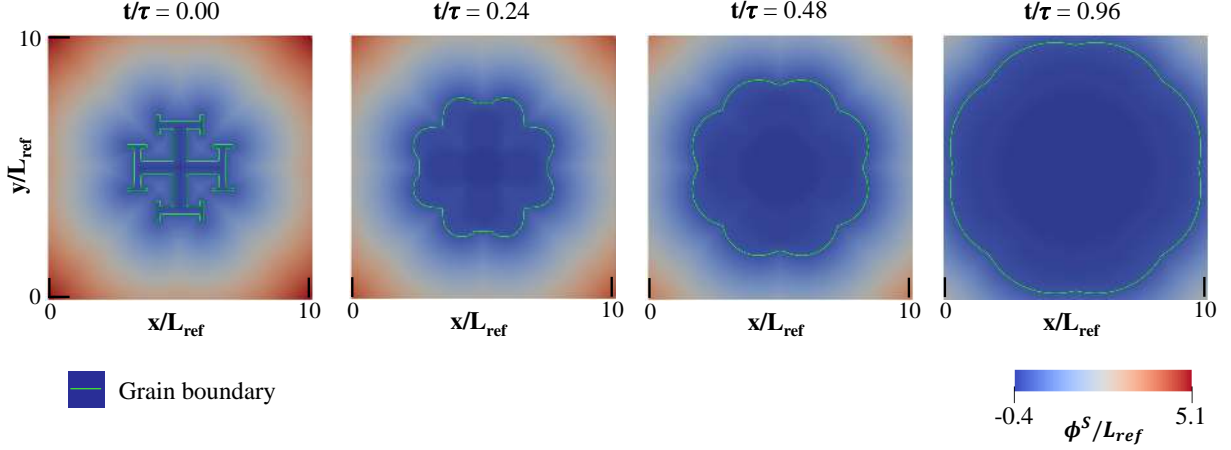
21

Figure 14: **Temporal evolution of signed wall distance field for a dendrite grain configuration. Green lines show the evolution of grain boundary.**
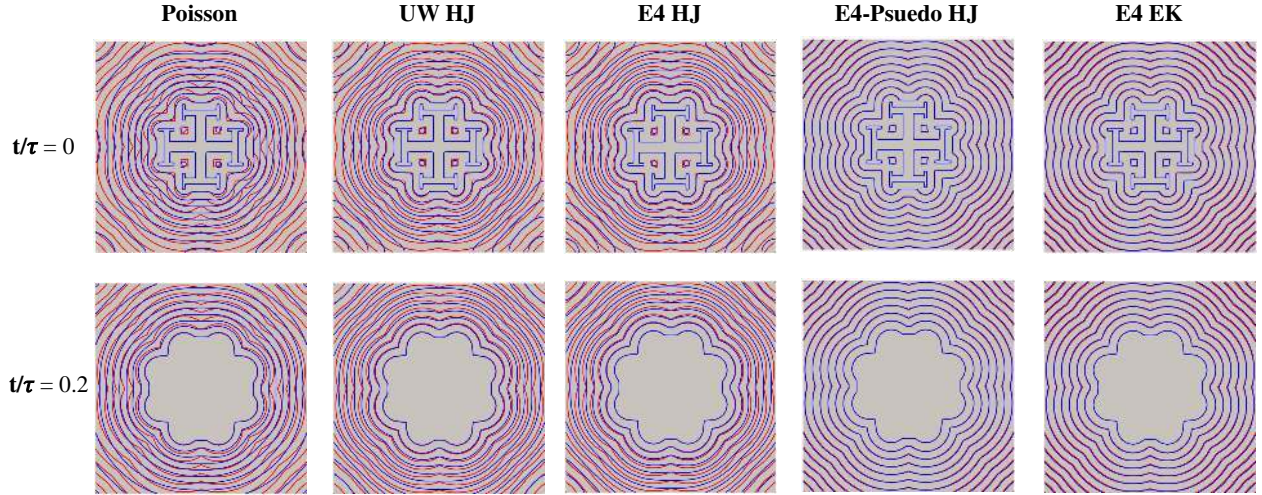

Figure 15: **Line contours of signed wall distance function at two time instances. Red lines correspond to the reference case where wall-distance field is initialized solving Eikonal equation using upwind schemes, and blue lines show wall-distance initialized using other schemes (Poisson/HJ/HJ LAD/ E4 Eikonal)**

It should however be noted that despite these differences in the level-set evolution, the grain-boundary can be accurately tracked using any of the aforementioned wall-distance based approaches as these are accurate close to the wall. To demonstrate this, iso-lines corresponding to different level-set values have been tracked in time. The grain is two-dimensional (extruding into the page) and hence the perimeter of level-set iso-lines is proportional to the surface area. Recall from Eq. 26 that the grain surface area in turn determines the chamber pressure. Figures 17 (b,c,d) show the temporal evolution of the perimeter for the level-sets of $\phi^s/L_{ref} = 0$, 1 and 2. Of these, $\phi^s/L_{ref} = 0$ represents the true grain boundary. Corresponding level-set iso-lines at $t/\tau = 0$ are shown in the inset plots for reference. Clearly, the wall-distance initialization procedure has little impact on the temporal evolution of the grain surface area. The typical two-peak evolution observed in pressure-time characteristic of a dendrite grain is captured. An

	Poisson	UW HJ	E4 HJ	E4-Psuedo HJ	E4 EK
$L_\infty =$	1.190	0.977	1.005	0.0529	0.055
$L_1 =$	2.97×10^{-2}	1.36×10^{-2}	1.25×10^{-2}	2.20×10^{-3}	2.96×10^{-3}

$t/\tau = 0$

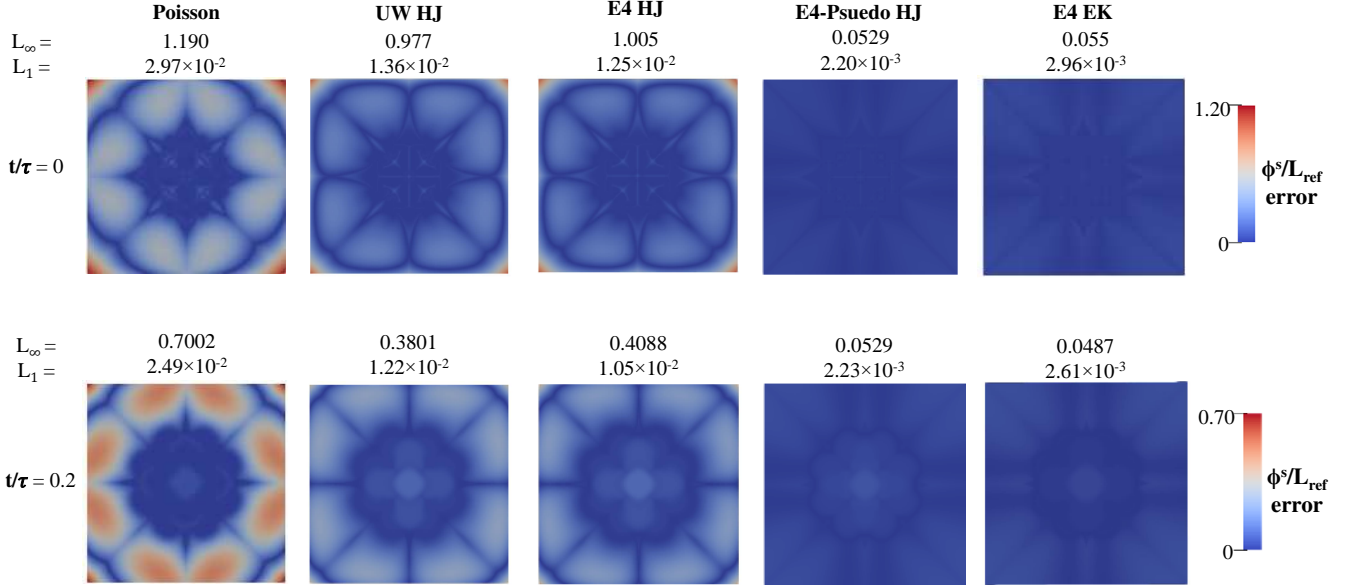|	Poisson	UW HJ	E4 HJ	E4-Psuedo HJ	E4 EK
$L_\infty =$	0.7002	0.3801	0.4088	0.0529	0.0487
$L_1 =$	2.49×10^{-2}	1.22×10^{-2}	1.05×10^{-2}	2.23×10^{-3}	2.61×10^{-3}

$t/\tau = 0.2$

Figure 16: **Absolute error contours of the signed wall distance function w.r.t the UW EK based initialization.**

initial drop in the surface area due to the burning of the dendrite projections is evident up to $t/\tau \approx 0.23$. This is followed by an increase in the surface area up to $t/\tau \approx 0.97$, beyond which it drops during the web burnout phase. Discrepancies due to the initial wall-distance field are observable for $\phi^s/L_{ref} = 1$ and 2; farther the level set higher the error. Consistent with the observations made earlier, predictions using E4-EK and E4-LAD are as accurate as UW-EK.

Apart from assisting in tracking the grain boundary, wall-distance field is also used in the turbulence models to predict the flow within the increasing chamber volume where the hot gases are generated (see figure 13 which shows wall-distance field enclosed within the grain boundary). Hence, the perimeter of a level set corresponding to $\phi^s/L_{ref} = -0.2$ is also tracked and is shown in 17 (a) demonstrating the sensitivity of the solution to the initial wall-distance field. It is worth noting that in typical CFD applications, it is sufficient to compute accurate wall distances close to the wall. On this front, even a Poisson equation yields acceptable results as illustrated in Figure 2(g). Nevertheless, accurate wall distances away from the wall are required in several other applications like evaluating the computational interface for overset grids [26], acoustic ray tracing inside jet mixing layers [27], etc. The current study demonstrates the computational efficacy of using LAD based HJ methods or Eikonal schemes using high-order schemes with filtering to yield accurate results in such applications.

Recommendations: The discussion so far has clearly brought out the benefits and disadvantages of different differential equation-based wall-distance based approaches (Poisson/Hamilton-Hacobi/Pseudo-HJ/Eikonal) using several spatial discretization schemes (Central schemes - explicit/compact and Upwind). The choice of a particular wall-distance approach depends on the application of interest. For applications where accurate wall-distances are required away from wall (CAD/automated meshing/computational interface for overset grids/acoustic ray tracing,etc), Eikonal/ LAD based HJ approach is preferable. On the other hand, standard HJ can be used for turbulence modelling applications as the Laplacian operator is tuned to exaggerate the wall-distance, $\phi > d_{true}$, near the fine convex features. We recommend explicit 4th order (E4) central difference scheme for solving either of these governing equations as it offers benefits
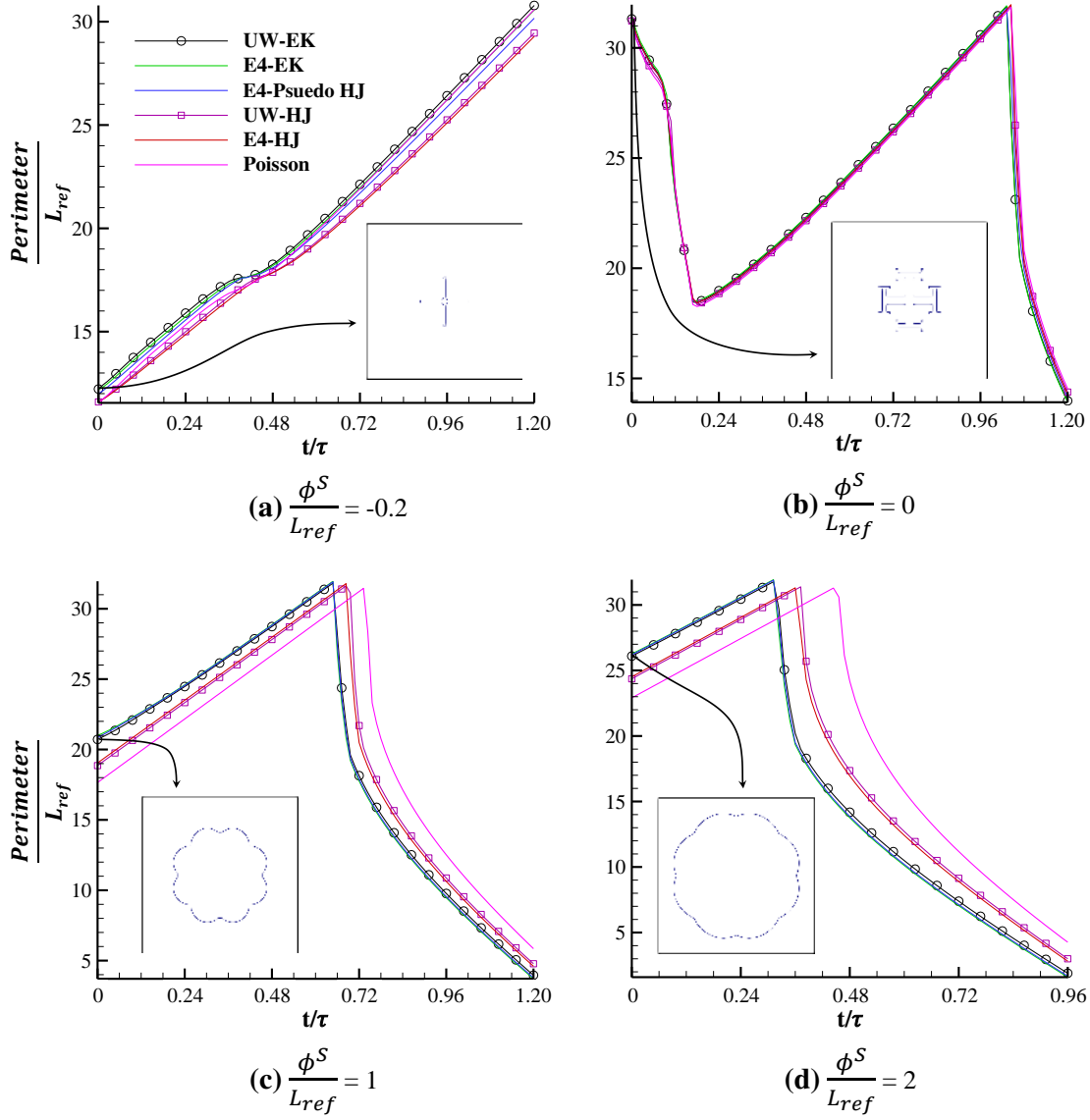
Figure 17: **Comparison of the temporal evolution of the grain perimeter with different initializations of the signed wall-distance field (obtained from different wall-distance approaches). (a-d) correspond to the iso-lines of different signed wall distance levels, ϕ^S/L_{ref}, as shown in the inset plots at $t/\tau = 0$.**

both in terms of speed and accuracy.

6. Conclusions

In this study, we have explored the efficacy of several differential equation based wall distance approaches (Poisson/Hamilton-Jacobi, HJ/Pseudo HJ/Eikonal) using both upwind and high order central difference (Explicit/Compact) schemes for spatial discretization. Firstly, the baseline solver which employs the commonly used upwind spatial discretization scheme is developed and validated on the canonical test cases. The computational benefit of using central difference schemes (E2, E4, C4 and C6) for wall-distance solvers, both in terms of computational time and accuracy, has been demonstrated. In addition, a modified

pseudo Hamilton-Jacobi (H-J) formulation based on Localized Artificial Diffusivity (LAD) approach has been proposed. A modified curvature correction has been implemented into the HJ solver to account for the error due to concave/convex wall curvatures. Using both the canonical and the unsteady test cases where the wall-distance evolves with time, the new pseudo HJ approach is demonstrated to be as accurate as the Eikonal approach. The unsteady test cases include 'piston-cylinder arrangement', 'bouncing cube' and wall-distance evolution inside the combustion chamber of a solid rocket motor due to the burning of a 'star grain propellant'.

While solving the Hamilton-Jacobi equation, the high-order central difference schemes performed approximately $1.4 - 2.8$ times faster than the upwind schemes with a marginal improvement in the solution accuracy. The HJ based on LAD scheme produced results which are of comparable accuracy to the Eikonal solution. It's solution accuracy is almost one order higher and the calculations are ≈ 1.5 times faster than the baseline HJ solver using upwind schemes. On the other hand, while solving the nonlinear hyperbolic Eikonal equation, the global error from high-order schemes is found to be larger when compared to the upwind schemes. Due to the lack of Laplacian operator in the Eikonal equation, the central schemes suffered with some dispersion errors despite filtering.

References

[1] Tucker, P. G., *Differential equation-based wall distance computation for DES and RANS*, Journal of computational physics. Volume 190, Issue 1, 229-248 (2003)

[2] Tucker, P. G., Rumsey, C. L., Spalart, P. R., Bartels, R. E., and Biedron, R. T., *Computations of wall distances based on differential equations*, AIAA journal, 43(3), 539-549. (2005)

[3] Richard J. Jefferson-Loveday, V. Nagabhushana Rao, James C. Tyacke and Paul G. Tucker, *High-order detached eddy simulation, zonal LES and URANS of cavity and labyrinth seal flows*, Int. J. Numer. Meth. Fluids. (2013)

[4] Tucker, P. G., *Hybrid Hamilton–Jacobi–Poisson wall distance function model*, Computers & fluids, 44(1), 130-142. (2011)

[5] Jing-lei XU, Chao YAN, Jing-jing FAN. *Computations of wall distances by solving a transport equation*, Appl. Math. Mech. -Engl. Ed., 32(2), 141–150. (2011)

[6] Anthony Bouchard, *Wall distance evaluation via Eikonal solver for RANS application*, Masters Thisis, UNIVERSITÉ DE MONTRÉAL. (2017)

[7] Miguel R. Visbal and Datta V. Gaitonde, *High-Order Schemes for Navier-Stokes Equations: Algorithm and Implementation Into FDL3DI*, AFRL. (1998)

[8] Jefferson-Loveday, R. J., Tucker, P. G., Northall, J. D., Nagabhushana Rao, V. *Differential equation specification of integral turbulence length scales*, Journal of turbomachinery, 135(3). (2013)

[9] E. Fares W. Schröder, *A differential equation for approximate wall distance*, Int. J. Numer. Meth. Fluids, Volume39, Issue8, (2002)

[10] Vadlamani, N. R., *Numerical investigation of separated flows in low pressure turbines*, Doctoral dissertation, University of Cambridge. (2014)

[11] Arnau Pons Lorente, *Study of grain burnback and performance of solid rocket motors.*, Project report, Universitat Politècnica de Catalunya. (2013)

[12] Spalart, P. R. and Allmaras, S. R. , *One-Equation Turbulence Model for Aerodynamic Flows*, La Recherche Aerospatiale, 1 (1994), pp. 5–21.

[13] Sebastian G., Charbel F., *Fast computation of the wall distance in unsteady Eulerian fluid-structure computations*, Int. J. Numer. Meth. Fluids, Volume 89, Issue 4-5, 143-161 (2019)

[14] Beatrice Roget, Jayanarayanan Sitaraman, *Wall distance search algorithm using voxelized marching spheres*, J. Comp Physics, Volume 241, 15 May 2013, Pages 76-94 (2013)

[15] Les A. Piegl, Wayne Tiller, *Algorithm for finding all k nearest neighbors*, Journal of Computer Aided Design, Volume 34, Issue 2, 82-172 (2002)

[16] Andrea Milli, Shahrokh Shahpar, *PADRAM: Parametric Design and Rapid Meshing System for Complex Turbomachinery Configurations*, ASME Turbo Expo, Copenhagen, Denmark (2012)

[17] Achu, Shankar, and Nagabhushana Rao Vadlamani, *Entropically Damped Artificial Compressibility Solver Using Higher Order Finite Difference Schemes on Curvilinear and Deforming Meshes.* In AIAA Scitech 2021 Forum, p. 0634. (2021)

[18] Vadlamani, Nagabhushana Rao, Paul G. Tucker, and Paul Durbin, *Distributed roughness effects on transitional and turbulent boundary layers*, Flow, Turbulence and Combustion 100, no. 3 p. 627-649. (2018)

[19] Lin, Y., Vadlamani, R., Savill, M. and Tucker, P., *Wall-resolved large eddy simulation for aeroengine aeroacoustic investigation*, The Aeronautical Journal, 121 (1242), pp.1032-1050. (2017)

[20] S. Kawai, S.K. Lele, *Localized artificial diffusivity scheme for discontinuity capturing on curvilinear meshes*, Journal of Computational Physics 227 (2008) 9498–9526

[21] Spalding DB. Calculation of turbulent heat transfer in cluttered spaces. Presented at the 10th international heat transfer conference, Brighton, UK; (1994).

[22] Tameo Nakanishi, *A Modification on Hamilton-Jacobi (HJ) Equation for Computing Wall Distances*, Trans. Japan Soc. Aero. Space Sci., Vol. 49, No. 163, pp. 55–57, (2006)

[23] Yildirim, Cengizhan and Aksel, Haluk, *Numerical simulation of the grain burnback in solid propellant rocket motor*, In 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, p. 4160 (2005).

[24] Hill, Philip G., and Carl R. Peterson. *Mechanics and thermodynamics of propulsion*, Reading (1992).

[25] Wei, R., Bao, F., Liu, Y., & Hui, W, *Robust three-dimensional level-set method for evolving fronts on complex unstructured meshes.* Mathematical Problems in Engineering, (2018).

[26] Nakahashi K, Togashi F, *Intergrid-boundary definition method for overset unstructured grid approach.* AIAA J 2000;38(11):2077–84.

[27] Xia, H., Tucker, P. G., & Dawes, W. N. *Level sets for CFD in aerospace engineering.* Progress in Aerospace Sciences, 46(7), 274-283 (2010).