Journal of Computational Physics, Volume 300, pp. 455-491, 2015
Available in a journal format at http://dx.doi.org/10.1016/j.jcp.2015.07.054

Improved second-order hyperbolic residual-distribution scheme and its extension to third-order on arbitrary triangular grids

Alireza Mazaheri*

NASA Langley Research Center, Hampton, VA 23681, USA.

Hiroaki Nishikawa

National Institute of Aerospace, Hampton, VA 23666, USA.

Abstract

In this paper, we construct second- and third-order hyperbolic residual-distribution schemes for general advection-diffusion problems on arbitrary triangular grids. We demonstrate that the accuracy of the second-order hyperbolic schemes in [J. Comput. Phys., 227 (2007) 315–352] and [J. Comput. Phys., 229 (2010) 3989–4016] can be greatly improved by requiring the scheme to preserve exact quadratic solutions. The improved second-order scheme can be easily extended to a third-order scheme by further requiring the exactness for cubic solutions. These schemes are constructed based on the SUPG methodology formulated in the framework of the residual-distribution method, and thus can be considered as economical and powerful alternatives to high-order finite-element methods. For both second- and third-order schemes, we construct a fully implicit solver by the exact residual Jacobian of the proposed second-order scheme, and demonstrate rapid convergence, typically with no more than 10–15 Newton iterations (and about 200–800 linear relaxations per Newton iteration), to reduce the residuals by ten orders of magnitude. We also demonstrate that these schemes can be constructed based on a separate treatment of the advective and diffusive terms, which paves the way for the construction of hyperbolic residual-distribution schemes for the compressible Navier-Stokes equations. Numerical results show that these schemes produce exceptionally accurate and smooth solution gradients on highly skewed and anisotropic triangular grids even for a curved boundary problem, without introducing curved elements. A quadratic reconstruction of the curved boundary normals and a high-order integration technique on curved boundaries are also provided in details.

Keywords: Higher-order, SUPG, Fluctuation splitting, Finite volume, Navier-Stokes, Quadratic reconstruction, Hermite polynomial, High-order integration

1. Introduction

In many flow simulations, accurate predictions of solution gradients, such as velocity and temperature gradients, are essential for design and analysis purposes as they are directly related to the physical quantities of interest: e.g., the viscous stresses, the vorticity, and the heat fluxes. However, it is widely accepted that accurate and smooth solution gradients cannot be achieved with conventional schemes on fully irregular unstructured grids [1, 2]. In conventional schemes, the gradients are obtained typically with a lower order of accuracy (e.g., through reconstruction of primary variables), and they are usually subject to numerical oscillations on such grids. The resolution of this issue is very important in justifying the use of high-fidelity models in engineering

*Corresponding author
Email address: ali.r.mazaheri@nasa.gov (Alireza Mazaheri)

design, analysis, and optimization, especially for applications involving complex geometries. The ability to predict the gradients on irregular grids is even more critical for grid adaptation, a vital technique for efficient CFD calculations in high-order methods [3], because the grid adaptation almost necessarily introduces irregularity in the grid. In fact, current practices in grid adaptation often avoid adaptation in certain regions such as within boundary layers where grid irregularity has a severe impact on the solution quality [4]. Therefore, numerical schemes that can accurately predict solution gradients on irregular grids need to be developed, so that the power of grid adaptation can be fully exploited.

There have been some efforts in the residual-distribution community in developing schemes that provide high-order solutions [5, 6, 7, 8, 9, 10], but little attention is paid, in general, in developing schemes that produce accurate and smooth high-order solution gradients. A good review of residual-distribution schemes is given in Ref. [11], in which a third-order residual-distribution scheme was presented for quadratic (P_2) elements. A different third-order residual-distribution scheme was also proposed and applied to RANS equations in Ref. [12], and for real-gas computations in Ref. [13]. These schemes produce no better than second-order accurate solution gradients, whether linear or quadratic elements are employed. A more improved third-order residual-distribution scheme was later proposed in Refs. [14, 15], and third-order accurate solution gradients were reported on some unstructured grids, including hybrid elements, using a proposed special gradient reconstruction strategy. However, the authors have noted that the third-order accurate solution gradients could not be reproduced for randomly distorted triangular elements. Quality of the predicted solution gradients using these schemes were not reported and therefore, are unknown.

The first-order hyperbolic system method (or the hyperbolic method for short), with which the solution and the solution gradients are simultaneously computed by solving a hyperbolic system for diffusion, provides a platform for construction of high-order schemes that could potentially produce high-order solution gradients that are both accurate and free of numerical noise.

The hyperbolic method was first studied for diffusion in Ref. [16] and then for advection-diffusion in Ref. [17] with Residual-Distribution (RD) schemes. Later, the method was demonstrated for the compressible Navier-Stokes equations by a second-order Finite-volume (FV) scheme [18]. Since then, there have been efforts in developing high-order hyperbolic schemes in the finite–volume method [19, 20, 21], in the active flux method [22], and in the RD method [23, 24] for unsteady computations.

In this paper, we focus on the development of hyperbolic RD schemes for two-dimensional problems, extending the previous work [16, 17], with several important advances.

- Improved Accuracy: We propose to construct a second-order scheme that preserves exact quadratic solutions, which can be accomplished by the curvature correction technique [8]. The resulting scheme remains compact for viscous problems, and produces significantly improved solution gradients over the previous schemes, which do not preserve exact quadratic solutions.

- Third-Order Accuracy: Extending the improved second-order scheme, we construct a third-order scheme that preserves exact cubic solutions. The construction requires quadratic least-squares (LSQ) gradients and a high-order source term discretization developed here.

- Nonlinear Equation: The improved schemes are extended to a nonlinear advection-diffusion equation by the preconditioned conservative formulation introduced in Ref. [18].

- High-Order on Linear Elements: We demonstrate that the third-order scheme does not require curved elements for curved boundary problems; it gives more accurate solution and gradients than the second-order scheme on the same linear grids (straight-sided elements). Our scheme is designed to be third-order accurate on straight-sided triangles, even for geometries containing curved boundaries. This is a significant advantage, because most high-order methods require curved geometries to be represented by high-order curved elements; see Ref. [3]. Our

proposed high-order scheme is not the only scheme that produces high-order solution on unstructured, straight-sided meshes. For example, the technique of Ref. [25], which was applied to Discontinuous Galerkin (DG) method, produces high-order accurate solution for geometries containing curved boundaries by locally approximating the curvature of the physical geometry (i.e., high-order normals) using information from the neighboring boundary elements (i.e., a local operation) with all triangles kept as straight-sided elements (not curved). The finite-volume (FV) scheme of Ref. [26] is another example, where a third-order solution was obtained on the linear elements with a quadratic reconstruction of the boundary normals for curved boundaries. The third-order residual-distribution schemes of Refs. [27, 8, 7], which are developed based on reconstruction techniques, are additional examples. Our proposed third-order scheme is among these schemes, and is more aligned with the FV scheme of Ref. [26] because the proposed third-order scheme here produces third-order solution gradients for geometries containing curved boundaries that are represented by straight-sided meshes.

- Non-Unified Approach: Instead of the fully integrated approach of discretizing the hyperbolic advection-diffusion system as in Ref. [17], we discretize the advective and diffusive terms separately. This approach will enable the extension to the compressible Navier-Stokes equations for which the eigenstructure of the whole system has not been discovered yet.

- Fully Implicit Solver: We construct a fully implicit solver for both second- and third-order schemes. For practical applications, explicit iterations considered in Refs. [16, 17] are not efficient enough, and a fully implicit solver is needed. The implicit solver is constructed by the exact residual Jacobian of the proposed second-order scheme. It converges typically in 10–15 Newton iterations and 200–800 Gauss-Seidel relaxations per Newton iteration to reduce the residual by 10 orders of magnitude for both second- and third-order schemes.

We demonstrate these features for a series of test problems involving fully irregular isotropic and anisotropic triangular grids and curved boundaries. This work serves as a basis for the development of high-order multidimensional hyperbolic RD schemes for more complex equations such as the Navier-Stokes equations. The extension of the proposed RD schemes to the Navier-Stokes equations will be addressed in a subsequent paper. Shocks and discontinuities will also be addressed in future work.

The paper is organized as follows. In the next section, we describe the basics of the hyperbolic RD scheme: residual evaluations, boundary conditions, and an implicit solver. In Section 3, we describe the construction of the SUPG distribution matrix based on a non-unified approach, in which the advective and diffusive terms are treated independently. In Section 4, we discuss the accuracy issue of the previous schemes, and propose a guiding principle for constructing improved schemes. A new second-order scheme is presented in Section 5, followed by the extension to third-order in Section 6. The extension to nonlinear advection-diffusion equations is explained in Section 7. Numerical results are then presented in Section 8 for both linear and nonlinear problems, followed by concluding remarks in Section 9.

2. Baseline RD scheme for advection-diffusion

In this section, we describe the baseline hyperbolic RD scheme [16] for the advection-diffusion equation on arbitrary triangular grids. This baseline scheme is the basis for the development of improved second-order and third-order schemes discussed later. For the purpose of our discussion and simplicity, we first describe the details for the linear advection-diffusion equation. Extension to the nonlinear equation is discussed in Sec. 7.

2.1. Hyperbolic advection-diffusion system

Consider a two-dimensional advection-diffusion equation:

$$\partial_t u + a\,\partial_x u + b\,\partial_y u = \nu\,(\partial_{xx} u + \partial_{yy} u) + \tilde{s}(x, y, u), \tag{1}$$

where a and b are constant advection speeds, respectively, in x and y directions, ν (> 0) is the diffusion coefficient, and $\tilde{s}(x, y, u)$ is a source term. Following Refs. [17, 23], we rewrite the above equation as a first-order hyperbolic advection-diffusion system:

$$\partial_\tau u + a\,\partial_x u + b\,\partial_y u = \nu\,(\partial_x p + \partial_y q) + s(x, y, u), \tag{2}$$

$$\partial_\tau p = (\partial_x u - p)/T_r, \tag{3}$$

$$\partial_\tau q = (\partial_y u - q)/T_r, \tag{4}$$

where τ is a pseudo time, $T_r > 0$ is the relaxation time, and $s(x, y, u)$ is a sum of $\tilde{s}(x, y, u)$ and a physical time-derivative term discretized by an implicit time-stepping scheme (see Refs. [23, 24] for more details). In the vector form, the system is written as

$$\frac{\partial \mathbf{U}}{\partial \tau} + \mathbf{A}\frac{\partial \mathbf{U}}{\partial x} + \mathbf{B}\frac{\partial \mathbf{U}}{\partial y} = \mathbf{Q}, \tag{5}$$

where

$$\mathbf{U} = \begin{bmatrix} u \\ p \\ q \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} a & -\nu & 0 \\ -1/T_r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b & 0 & -\nu \\ 0 & 0 & 0 \\ -1/T_r & 0 & 0 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} s \\ -p/T_r \\ -q/T_r \end{bmatrix}. \tag{6}$$

The system is hyperbolic in the pseudo time as shown in Ref. [17]. Towards the pseudo steady state, the variables p and q approach the solution gradients u_x and u_y and hence the above equation becomes identical to the original advection-diffusion equation with the physical time-derivative discretized by an implicit time-stepping scheme(see Ref. [23]). This is true for any T_r, and thus T_r is determined not by physical constraints but by optimal steady convergence criteria [17], leading to a non-stiff hyperbolic formulation for diffusion. Note that simply dropping the pseudo time derivative will also recover the original equation. In this paper, we focus on steady problems, but the resulting steady schemes can be made time accurate by including a physical time derivative term in $s(x, y, u)$ as described in Refs. [23, 24]. In either case, an efficient steady solver is required, and its development is one of the objectives of the present paper.

2.2. RD scheme: cell-residual, distribution matrix, nodal residual

We discretize the hyperbolic advection-diffusion system on unstructured triangular grids. The domain is divided into a set $\{E\}$ of arbitrary triangular cells (or elements), and an associated set $\{J\}$ of nodes (or vertices). The total number of nodes is denoted by N. We store the solutions (u_j, p_j, q_j) at each node $j \in \{J\}$. In the RD method, we first evaluate the residuals over the elements and then distribute the residuals to the nodes with a distribution matrix. The cell-residual over an

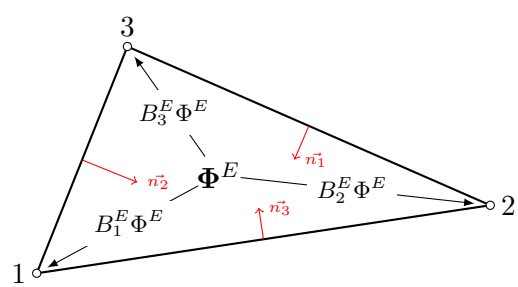


Figure 1: Schematic of the residual distribution to local nodes and definition of the inward unit normals (not-to-scale).

element $E \in \{E\}$ is defined as

$$\mathbf{\Phi}^E = \begin{bmatrix} \Phi_u^E \\ \Phi_p^E \\ \Phi_q^E \end{bmatrix} = \int_E (-\mathbf{A}\mathbf{U}_x - \mathbf{B}\mathbf{U}_y + \mathbf{Q})dxdy. \tag{7}$$

We assume a piecewise linear variation of $\mathbf{U}$ over the element, which interpolates the three nodal solutions, and perform the integration to obtain

$$\mathbf{\Phi}^E = -\sum_{i=1}^{3} \mathbf{K}_i \mathbf{U}_i + \overline{\mathbf{Q}}^E d\Omega^E, \tag{8}$$

where i is the local vertex counter (i.e., $i = 1, 2, 3$) for the element E, $d\Omega^E$ is the element area, and

$$\mathbf{K}_i = \frac{1}{2}(\mathbf{A}\,\hat{n}_{i_x} + \mathbf{B}\,\hat{n}_{i_y})|\mathbf{n}_i| = \frac{1}{2}\mathbf{A}_{n_i}|\mathbf{n}_i|, \quad \overline{\mathbf{Q}}^T = \tfrac{1}{3}\sum_{i=1}^{3} \mathbf{Q}_i. \tag{9}$$

Note that $\mathbf{n}_i = (n_{i_x}, n_{i_y})$ is the scaled inward normal to the face (edge) opposed to the vertex i, $|\mathbf{n}_i| = \sqrt{n_{i_x}^2 + n_{i_y}^2}$ (see Fig. 1), and $\hat{\mathbf{n}}_i = (\hat{n}_{i_x}, \hat{n}_{i_y})$ is the unit inward normal. We note that for every element $\sum_{i=1}^{3} \mathbf{n}_i = 0$ and therefore we have $\sum_{i=1}^{3} \mathbf{K}_i = 0$.

The next step is to distribute them to the three vertices by the distribution matrices, $\mathbf{B}_1^E$, $\mathbf{B}_2^E$, $\mathbf{B}_3^E$, as illustrated in Fig. 1. The discussion of the distribution matrix is one of the key contributions of the present paper, but we leave the choice open at this point. We only mention here that $\mathbf{B}_i^E$ is a 3×3 matrix, which sums up to the identity matrix over the element for conservation. The distribution process results in the nodal residual at node j:

$$\mathbf{Res}_j = \frac{1}{d\Omega_j} \sum_{E \in \{E_j\}} \mathbf{B}_j^E \mathbf{\Phi}_j^E, \tag{10}$$

where $d\Omega_j$ is the median dual volume around the node j (see Fig. 2), and $\{E_j\}$ is a set of triangular elements sharing the node j. The nodal residual is an approximation of the spatial part of the target equations, and thus it leads to a semi-discrete form:

$$\frac{d\mathbf{U}_j}{d\tau} = \mathbf{Res}_j. \tag{11}$$

It can be integrated to the steady state by explicit pseudo–time stepping schemes as in Refs. [16, 17], which are significantly faster than conventional explicit schemes because the hyperbolic formulation eliminates a typical diffusion constraint, $O(h^2)$, where h is a representative mesh spacing, on the explicit time step [16, 17]. However, it still requires a large number of iterations, especially for anisotropic grids. To improve the convergence, we developed an implicit solver in this work.

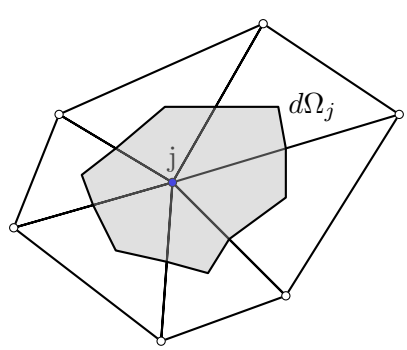


Figure 2: Schematic of a median dual volume around the node j.

2.3. Implicit solver

To solve Eq. (11) for the pseudo steady state, we drop the pseudo-time derivative, and define the global system of steady residual equations,

$$0 = \mathbf{Res}, \tag{12}$$

which consists of the right hand side of Eq. (11) for all nodes. Note that the resulting residual equation has become consistent with the advection-diffusion equation (1) because the pseudo-time derivative has been dropped. We solve the system by Newton's method:

$$\mathbf{U}^{l+1} = \mathbf{U}^l + \Delta\mathbf{U}^l, \tag{13}$$

where $\mathbf{U} = (u_1, p_1, q_1, u_2, p_2, q_2, \ldots, u_N, p_N, q_N)$, and l is the iteration counter. The correction $\Delta\mathbf{U}^l = \mathbf{U}^{l+1} - \mathbf{U}^l$ is determined as the solution to the linear system:

$$-\frac{\partial \mathbf{Res}}{\partial \mathbf{U}}\Delta\mathbf{U}^l = \mathbf{Res}^l(\mathbf{U}^l), \tag{14}$$

where $\mathbf{Res}^l$ is the steady residual vector evaluated by $\mathbf{U}^l$. The Jacobian matrix $\partial\mathbf{Res}/\partial\mathbf{U}$ is exact and sparse because the spatial discretization is compact. For each node $j \in \{J\}$, it involves $(k+1)$ 3×3 blocks, where k is the number of immediate neighboring nodes to the node j. For example, $k = 6$ for the node j shown in Fig. 2 and therefore, for this particular node, there are seven 3×3 blocks. For a node j, we have

$$-\mathbf{J}_j\Delta\mathbf{U}_j^l - \sum_{m=1}^{k}\mathbf{J}_{j,m}\Delta\mathbf{U}_m^l = \mathbf{Res}_j(\mathbf{U}^l), \tag{15}$$

where

$$\mathbf{J}_j = \frac{\partial \mathbf{Res}_j}{\partial \mathbf{U}_j}, \quad \mathbf{J}_{j,m} = \frac{\partial \mathbf{Res}_j}{\partial \mathbf{U}_m}. \tag{16}$$

We may analytically evaluate the diagonal and off-diagonal entries of the Jacobian matrix, i.e., $\mathbf{J}_j$ and $\mathbf{J}_{j,m}$, but this process is rather tedious for general time–dependent advection-diffusion problems, and even more difficult for complex systems such as the Navier-Stokes equations. To overcome the difficulty, we implemented an *Automatic Differentiation* (AD) tool based on an operator-overloading technique to evaluate the exact Jacobians numerically through chain rule. Thus, the Jacobian matrix is exact up to the round-off error for the baseline and second-order schemes. The linear system is relaxed by the sequential Gauss-Seidel relaxation with an under-relaxation parameter introduced for stabilization (see Ref. [28]). Typically, the relaxation is performed until the linear residual is reduced by five orders of magnitude with a maximum relaxation steps of 1000.

2.4. Implicit boundary condition

Here, we discuss two Boundary Condition (BC) types: Dirichlet (i.e., u is known), and Neumann (i.e., $\partial_n u$ is known). We note that because gradients are also the primary variables in the hyperbolic method (i.e., not reconstructed from the solution variable), the Neumann type BC is treated similarly as the Dirichlet type BC.

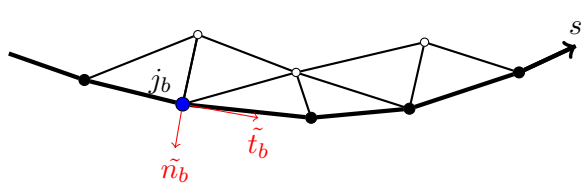


Figure 3: Schematic of boundary nodes for formulation of implicit boundary condition.

For a Dirichlet type BC, the following set of equations is imposed at the boundary nodes:

$$u - u_b = 0, \tag{17}$$

$$(p, q)\cdot\tilde{t}_b = \partial_s u_b, \tag{18}$$

$$(\partial_x u - p, \partial_y u - q)\cdot\tilde{n}_b = 0, \tag{19}$$

where u_b is the given boundary value and $\partial_s u_b$ is the derivative of the u_b along the boundary, which can be computed with the given u_b. The $\tilde{n}_b$ and $\tilde{t}_b$ are, respectively, the unit normal and tangent vectors at the boundary nodes (see Fig. 3). For a strong Dirichlet type BC, we specify the BC value at the boundary nodes and solve the hyperbolic system accordingly. Here, we discretize and solve Eq. (19) according to the hyperbolic RD scheme described in the previous section. That is, the residual vector at the node j_b is redefined as

$$\begin{bmatrix} u_{j_b} - u_b \\ (p_{j_b}, q_{j_b}) \cdot \tilde{t}_b - \partial_s u_b \\ \mathbf{Res}_{j_b}(2)\tilde{n}_{b_x} + \mathbf{Res}_{j_b}(3)\tilde{n}_{b_y} \end{bmatrix}, \tag{20}$$

where $\mathbf{Res}_{j_b}(2)$ and $\mathbf{Res}_{j_b}(3)$ are the nodal residuals of the second and third equations of the first-order hyperbolic system (i.e., Eq. 3 and Eq. 4, respectively), computed as described in the previous sections. Alternatively, we can only specify the solution value on the boundaries and compute the solution gradients using the proposed schemes

$$\begin{bmatrix} u_{j_b} - u_b \\ \mathbf{Res}_{j_b}(2) \\ \mathbf{Res}_{j_b}(3) \end{bmatrix}, \tag{21}$$

which is more convenient than the previous approach, Eq. (20), because it neither needs the boundary tangent vector nor the boundary tangent derivatives. We have implemented both approaches (i.e., Eqs. 20 and 21) and similar results are obtained with the examples presented in the Sec. 8. The Dirichlet type BC is relevant to the no-slip boundary condition for the Navier-Stokes equations.

For a Neumann type BC, we impose the following equations to the boundary nodes:

$$\partial_\tau u + a\partial_x u + b\partial_y u = \nu(\partial_x p + \partial_y q) + s, \tag{22}$$

$$(p, q) \cdot \tilde{n}_b = \partial_n u_b, \tag{23}$$

$$(\partial_x u - p, \partial_y u - q) \cdot \tilde{t}_b = 0, \tag{24}$$

where $\partial_n u_b$ is the given gradient of the u on the boundary in the normal direction, and therefore is known. For the strong formulation of a Neumann type BC, we discretize and solve Eqs. (22) and (24) according to the RD scheme; that is

$$\begin{bmatrix} \mathbf{Res}_{j_b}(1) \\ (p_{j_b}, q_{j_b}) \cdot \tilde{n}_b - \partial_n u_b \\ \mathbf{Res}_{j_b}(2)\tilde{t}_{b_x} + \mathbf{Res}_{j_b}(3)\tilde{t}_{b_y} \end{bmatrix}, \tag{25}$$

where $\mathbf{Res}_{j_b}(1)$ is the nodal residual of the first equation of the first-order hyperbolic system computed as described in the previous sections. An example of the Neumann type BC is the adiabatic condition used in the Navier-Stokes equations.

In either of the Dirichlet or the Neumann type BC, the modified residual is incorporated into the implicit solver with the exact Jacobians. In doing so, special care must be taken to avoid encountering zero diagonal entries in the Jacobian of the boundary nodes. To examine this, let us consider a Neumann type BC, Eq. (25), and a boundary node whose normals defined as $\tilde{n}_b = (0, 1)$. For this specific boundary node, the diagonal entry of the Jacobian, corresponding to the second equation, becomes zero; i.e.,

$$\frac{\partial}{\partial p_{j_b}} \left((p_{j_b}, q_{j_b}) \cdot \tilde{n}_b - \partial_n u_b \right) = 0,$$

and therefore, the linear relaxation fails. To remedy this, we simply swap the second and third equations for the corresponding boundary nodes, according to their magnitudes of $\tilde{n}_{b_x}$ and $\tilde{n}_{b_y}$; that is,

$$\begin{bmatrix} \mathbf{Res}_{j_b}(1) \\ \mathbf{Res}_{j_b}(2)\tilde{t}_{b_x} + \mathbf{Res}_{j_b}(3)\tilde{t}_{b_y} \\ (p_{j_b}, q_{j_b}) \cdot \tilde{n}_b - \partial_n u_b \end{bmatrix}, \quad \text{if} \quad |\tilde{n}_{b_x}| \leq |\tilde{n}_{b_y}|, \tag{26}$$

otherwise, we use Eq. (25). With this technique, the Jacobian of the boundary nodes is always invertible. This technique is further illustrated in Sec. 8.2.2 through an example.

2.5. Remark on variable advection coefficients

If the advection coefficients a and b are functions of (x, y), advective fluxes are not necessarily conservative. In this case, we estimate the advection vector by the arithmetic average over an element to write

$$\frac{\partial \mathbf{U}}{\partial \tau} + \overline{\mathbf{A}}\frac{\partial \mathbf{U}}{\partial x} + \overline{\mathbf{B}}\frac{\partial \mathbf{U}}{\partial y} = \mathbf{Q}, \tag{27}$$

where $\overline{\mathbf{A}}$ and $\overline{\mathbf{B}}$ are defined with the averaged advection vector $(\overline{a}, \overline{b})$:

$$\mathbf{A} = \begin{bmatrix} \overline{a} & -\nu & 0 \\ -1/T_r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \overline{b} & 0 & -\nu \\ 0 & 0 & 0 \\ -1/T_r & 0 & 0 \end{bmatrix}. \tag{28}$$

The baseline RD scheme is then applicable. If conservative fluxes exist, then the nonlinear formulation described in Sec. 7 can be employed to construct an RD scheme.

3. Distribution matrices: non-unified approach

The RD scheme is defined by the combination of the cell-residuals and the distribution matrix. The distribution matrix is mainly responsible for the dissipative behavior and the stability while the cell-residuals determine the order of accuracy. Unbounded distribution matrices can also affect the order of accuracy, but we consider only bounded ones in this paper. The construction of the distribution matrix often requires the knowledge of the eigenstructure of the target system. The eigenstructure of the hyperbolic advection-diffusion system is known [17] and various distribution matrices can be constructed. However, the extension to the compressible Navier-Stokes equations is not possible at the moment because the eigenstructure has not been successfully worked out yet. In order to develop hyperbolic RD schemes that can be extended to the compressible Navier-Stokes equations, we consider a non-unified approach introduced for FV schemes in Refs. [18] and [20], respectively, for the compressible Navier-Stokes and advection-diffusion equations. In this approach, we construct distribution matrices for advection and diffusion separately and then combine them to derive a distribution scheme for the advection-diffusion equation. This approach easily extends to the compressible Navier-Stokes equations because the eigenstructures of the inviscid terms and the hyperbolized viscous terms can be fully analyzed independently [18].

3.1. Unified approach

To illustrate the point of the non-unified approach, we first recall the unified approach of Ref. [17]. Consider an arbitrary unit vector $\hat{\mathbf{n}} = (\hat{n}_x, \hat{n}_y)$. The unified advection-diffusion flux Jacobian is defined as:

$$\mathbf{A}_n = \mathbf{A}\hat{n}_x + \mathbf{B}\hat{n}_y = \begin{bmatrix} a_n & -\nu\,\hat{n}_x & -\nu\,\hat{n}_y \\ -\hat{n}_x/T_r & 0 & 0 \\ -\hat{n}_y/T_r & 0 & 0 \end{bmatrix}, \tag{29}$$

where $a_n = a\,\hat{n}_x + b\,\hat{n}_y$ is the advection velocity projected onto the unit vector $\hat{\mathbf{n}}$. The Jacobian $\mathbf{A}_n$ has the following real eigenvalues:

$$\lambda_1 = \frac{1}{2}\left[a_n - \sqrt{a_n^2 + \frac{4\nu}{T_r}}\,\right], \quad \lambda_2 = \frac{1}{2}\left[a_n + \sqrt{a_n^2 + \frac{4\nu}{T_r}}\,\right], \quad \lambda_3 = 0, \tag{30}$$

where $\lambda_1 < 0$ and $\lambda_2 > 0$. The arbitrary but positive relaxation time, T_r, is defined as

$$T_r = \frac{L_r}{\sqrt{a^2 + b^2} + \nu/L_r}, \tag{31}$$

where the length scale L_r is a quantity of $O(1)$, which may be taken as $L_r = 1/2\pi$, as used here, or from an optimal formula as derived in Ref. [17]. The right and left eigenvectors for the Jacobian $\mathbf{A}_n$ are given by

$$\mathbf{R}_n = \begin{bmatrix} -\lambda_1 T_r & -\lambda_2 T_r & 0 \\ \hat{n}_x & \hat{n}_x & -\hat{n}_y \\ \hat{n}_y & \hat{n}_y & \hat{n}_x \end{bmatrix}, \quad \mathbf{L}_n = \tfrac{1}{\lambda_1 - \lambda_2} \begin{bmatrix} -1/T_r & -\lambda_2 \hat{n}_x & -\lambda_2 \hat{n}_y \\ 1/T_r & \lambda_1 \hat{n}_x & \lambda_1 \hat{n}_y \\ 0 & -(\lambda_1 - \lambda_2)\hat{n}_y & (\lambda_1 - \lambda_2)\hat{n}_x \end{bmatrix}. \tag{32}$$

The unified Jacobian $\mathbf{A}_n$ can be rewritten based on its right and left eigenvectors as

$$\mathbf{A}_n = \mathbf{R}_n \mathbf{\Lambda}_n \mathbf{L}_n, \tag{33}$$

where $\mathbf{\Lambda}$ is the diagonal matrix with the entries of the eigenvalues defined in Eq. (30). The Jacobian matrix can be decomposed as (see Ref. [17])

$$\mathbf{A}_n = \lambda_1 \mathbf{\Pi}_1 + \lambda_2 \mathbf{\Pi}_2 = \mathbf{A}_n^- + \mathbf{A}_n^+, \tag{34}$$

where

$$\mathbf{\Pi}_1 = \frac{1}{\lambda_1 - \lambda_2} \begin{bmatrix} \lambda_1 & \lambda_1\lambda_2 T_r \hat{n}_x & \lambda_1\lambda_2 T_r \hat{n}_y \\ -\hat{n}_x/T_r & -\lambda_2 \hat{n}_x^2 & -\lambda_2 \hat{n}_x \hat{n}_y \\ -\hat{n}_y/T_r & -\lambda_2 \hat{n}_x \hat{n}_y & -\lambda_2 \hat{n}_y^2 \end{bmatrix}, \tag{35}$$

$$\mathbf{\Pi}_2 = \frac{1}{\lambda_1 - \lambda_2} \begin{bmatrix} -\lambda_2 & -\lambda_1\lambda_2 T_r \hat{n}_x & -\lambda_1\lambda_2 T_r \hat{n}_y \\ \hat{n}_x/T_r & \lambda_1 \hat{n}_x^2 & \lambda_1 \hat{n}_x \hat{n}_y \\ \hat{n}_y/T_r & \lambda_1 \hat{n}_x \hat{n}_y & \lambda_1 \hat{n}_y^2 \end{bmatrix}. \tag{36}$$

Thus, the $\mathbf{K}_i$ defined in Eq. (9) can be written as

$$\mathbf{K}_i = \mathbf{K}_i^- + \mathbf{K}_i^+, \tag{37}$$

where

$$\mathbf{K}_i^- = \frac{1}{2}\lambda_{1,i}\,\mathbf{\Pi}_{1,i}\,|\mathbf{n}_i| = \frac{1}{2}\mathbf{A}_{n_i}^-|\mathbf{n}_i|, \tag{38}$$

$$\mathbf{K}_i^+ = \frac{1}{2}\lambda_{2,i}\,\mathbf{\Pi}_{2,i}\,|\mathbf{n}_i| = \frac{1}{2}\mathbf{A}_{n_i}^+|\mathbf{n}_i|. \tag{39}$$

These matrices are often required for the construction of the distribution matrix, and as can be seen from the above, they require the eigenvalues and the eigenvectors, which are not known at present for a hyperbolic formulation of the compressible Navier-Stokes equations.

3.2. Non-unified approach

In the non-unified approach [18, 20], we treat the advective and diffusive terms separately. Therefore we rewrite Eq. (5) as

$$\frac{\partial \mathbf{U}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \mathbf{Q}, \tag{40}$$

where

$$\mathbf{F} = \mathbf{F}^a + \mathbf{F}^d = \begin{bmatrix} au \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -\nu p \\ -u/T_r \\ 0 \end{bmatrix}, \tag{41}$$

$$\mathbf{G} = \mathbf{G}^a + \mathbf{G}^d = \begin{bmatrix} bu \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -\nu q \\ 0 \\ -u/T_r \end{bmatrix}. \tag{42}$$

Similarly, the flux Jacobian $\mathbf{A}_n$ is decomposed into two advective and diffusive fluxes, $\mathbf{A}_n^a$ and $\mathbf{A}_n^d$, respectively:

$$\mathbf{A}_n = \mathbf{A}_n^a + \mathbf{A}_n^d, \tag{43}$$

where

$$\mathbf{A}_n^a = \begin{bmatrix} a_n & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{A}_n^d = \begin{bmatrix} 0 & -\nu\hat{n}_x & -\nu\hat{n}_y \\ -\hat{n}_x/T_r & 0 & 0 \\ -\hat{n}_y/T_r & 0 & 0 \end{bmatrix}. \tag{44}$$

The distribution matrix is constructed separately for the advective and diffusive terms based on the corresponding flux Jacobian, and then the two matrices are combined to form a matrix for the advection-diffusion equation. This approach can be extended to the compressible Navier-Stokes equations because the eigenstructure of the hyperbolized viscous terms is available [18].

The advective Jacobian has only one non-zero eigenvalue ($\lambda^a = a_n$) and thus we can simply decompose it according to the sign of λ^a:

$$\mathbf{A}_n^a = \mathbf{A}_n^{a^+} + \mathbf{A}_n^{a^-} = \begin{bmatrix} \max(0, a_n) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} \min(a_n, 0) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{45}$$

The diffusive Jacobian has two non-zero eigenvalues, $\lambda_1^d = -\sqrt{\nu/T_r}$, $\lambda_2^d = +\sqrt{\nu/T_r}$, with the right and left eigenvectors formally in the form of Eq. (32). Note, however, that they are numerically different because the eigenvalues are different. The diffusive Jacobian can be split as

$$\mathbf{A}_n^d = \mathbf{R}_n^d \mathbf{\Lambda}_n^d \mathbf{L}_n^d = \lambda_1^d \mathbf{\Pi}_1 + \lambda_2^d \mathbf{\Pi}_2 = \mathbf{A}_n^{d^+} + \mathbf{A}_n^{d^-}, \tag{46}$$

where $\mathbf{\Pi}_1$ and $\mathbf{\Pi}_2$ are given by Eq. (35) and Eq. (36), respectively, with $\lambda_1 = \lambda_1^d$ and $\lambda_2 = \lambda_2^d$.

We define $\mathbf{K}_i^d$ by

$$\mathbf{K}_i^d = \frac{1}{2}\mathbf{A}_{n_i}^d |\mathbf{n}_i| = \mathbf{K}_i^{d^-} + \mathbf{K}_i^{d^+}, \tag{47}$$

where the negative and positive matrices, $\mathbf{K}_i^{d^-}$ and $\mathbf{K}_i^{d^+}$ are given by

$$\mathbf{K}_i^{d^-} = \frac{1}{2}\lambda_{1,i}^d \mathbf{\Pi}_{1,i} |\mathbf{n}_i| = \frac{1}{2}\mathbf{A}_{n_i}^{d^-} |\mathbf{n}_i|, \tag{48}$$

$$\mathbf{K}_i^{d^+} = \frac{1}{2}\lambda_{2,i}^d \mathbf{\Pi}_{2,i} |\mathbf{n}_i| = \frac{1}{2}\mathbf{A}_{n_i}^{d^+} |\mathbf{n}_i|. \tag{49}$$

In the following section, we describe the the Streamline-Upwind-Petrov-Galerkin (SUPG) distribution matrix and its formulation in the context of hyperbolic RD schemes. For completeness and comparison purposes, the SUPG distribution matrix presented for both unified and non-unified approaches.

3.3. Distribution matrix: Streamline-Upwind-Petrov-Galerkin (SUPG)

Various distribution matrices can be employed for the hyperbolic RD schemes: Lax-Wendroff [16], LDA [17], and SUPG distribution matrices. However, in our experience, hyperbolic RD schemes with the LDA distribution matrix, although it can be constructed by the non-unified approach, have an instability problem in the linear relaxation on some anisotropic triangular grids. On the other hand, hyperbolic RD schemes can be successfully stabilized with a suitable under-relaxation parameter in the case of the SUPG distribution matrix. Therefore, in this paper, we focus on the SUPG distribution matrix. The finite-element SUPG scheme [29] is known to be applicable to the RD framework [30].

In the unified approach, the SUPG distribution matrix for triangular grids is given by

$$\tilde{\mathbf{B}}_i^{\mathrm{SUPG}} = \frac{1}{3}\mathbf{I} + \frac{1}{2}\mathbf{K}_i \left(\sum_{l=1}^{3} \mathbf{K}_l^{+} \right)^{-1}, \tag{50}$$

which consists of the Galerkin and the stabilization terms. In the non-unified approach, we construct the SUPG distribution matrix as follows:

$$\mathbf{B}_i^{\mathrm{SUPG}} = \frac{1}{3}\mathbf{I} + \mathbf{D}_i^a + \mathbf{D}_i^d, \tag{51}$$

where $\mathbf{D}_i^a$ and $\mathbf{D}_i^d$ are the stabilization terms defined independently for the advective and diffusive terms,

$$\mathbf{D}_i^a = \begin{bmatrix} d_i^{\mathrm{SUPG}} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad d_i^{\mathrm{SUPG}} = \frac{1}{2} \frac{a_{n_i}|n_i|}{\sum_{l=1}^{3} \max(0, a_{n_l})|n_l|}, \tag{52}$$

$$\mathbf{D}_i^d = \frac{1}{2}\mathbf{K}_i^d \left(\sum_{l=1}^{3} \mathbf{K}_l^{d+} \right)^{-1}. \tag{53}$$

Note that the denominator of Eq. (52) cannot vanish unless the advection vector is exactly zero. A small numerical value of the order of machine zero may be added to the denominator in order to avoid zero division in the case of zero advection vector. Another possibility is to completely remove D_i^a term for zero advection vector; both approaches yield identical results.

It is also important to note that we have $\sum_{i=1}^{3} \mathbf{D}_i^a = \sum_{i=1}^{3} \mathbf{D}_i^d = 0$, and therefore the conservation property is guaranteed:

$$\sum_{i=1}^{3} \mathbf{B}_i^{\mathrm{SUPG}} = \mathbf{I}. \tag{54}$$

A weighting function can be introduced in the stabilization terms of Eq. (51),

$$\mathbf{B}_i^{\mathrm{SUPG}} = \frac{1}{3}\mathbf{I} + (1-\omega)\mathbf{D}_i^a + \omega \mathbf{D}_i^d, \tag{55}$$

where ω is defined such that $\omega \to 0$ as $Re \to \infty$, and $\omega \to 1$ as $Re \to 0$, so that the scheme will properly reduce to pure advection and diffusion schemes as $Re \to \infty$ and $Re \to 0$, respectively. The weighting function ω can be evaluated, based on a detailed analysis for a one-dimensional hyperbolic advection-diffusion system [17], as

$$\omega = \frac{2}{Re + 2}, \tag{56}$$

where we define the Re for the two-dimensional system as

$$Re = \sqrt{a^2 + b^2}/\nu. \tag{57}$$

In our study, however, we observed no noticeable change in the results between Eq. (51) and Eq. (55) with the above weighting function. Therefore, the results shown in this paper are all based on the SUPG distribution matrix without the weighting function. Details of the effects of the weighting function are a subject for future study.

As can be seen from above, the SUPG distribution matrix is constructed based on a decomposition of the stabilization term into advective and diffusive stabilization terms. Therefore, the SUPG distribution matrix can be extended to a hyperbolic formulation of the compressible Navier-Stokes equations, for which the inviscid and viscous terms can be analyzed independently [18].

4. New design principle for advection-diffusion equation

An important feature of the RD schemes critical to the accuracy condition is the ability to preserve exact polynomial solutions on arbitrary grids. Historically, RD schemes have been designed to preserve linear exact solutions for hyperbolic systems of conservation laws. This is called the linearity preservation. Such schemes are known to yield second-order discretization errors. The baseline hyperbolic RD scheme [16] is constructed based on the linearity preserving property. Although it gives second-order accuracy for all variables on regular triangular grids [16], it was found later that the order of accuracy of the gradient variables (i.e., p and q) deteriorates to first-order on irregular grids. In a subsequent work presented in Ref. [17], a better scheme was proposed that attempted to improve the accuracy of the gradients by upgrading the evaluation of the second and third components of the residual. In this paper, the scheme of Ref. [17] is referred to as the RD-JCP2010 scheme. Results given in Ref. [17] showed that the RD-JCP2010 scheme, relative to the baseline scheme, improves the order of accuracy of the gradient variables, but no results were presented for the quality of the predicted gradients. As will be shown later in this paper, we found that for a different test problem and a more general exact solution, the same RD-JCP2010 scheme does not yield second-order accuracy for the gradient variables (i.e., p and q), and also generates severe oscillations in these variables within the domain. It appears that the discretization of the source terms (including those arising from the hyperbolic formulation) was not fully compatible with the flux terms, and it is not clear to us, at the moment, how to improve the source term discretization to develop a fully second-order scheme. To overcome this difficult problem, we instead propose to improve the flux terms such that the residual vanishes for quadratic solutions for the advection-diffusion equation. The exactness, as will be demonstrated later, has a significant impact on the accuracy and the quality of the gradient variables on irregular grids. The construction of such schemes is very difficult in general because second-order gradients need to be recovered from second-order accurate solutions on irregular grids for the diffusive term. However, the construction is quite straightforward in the hyperbolic method since the gradients are computed simultaneously with the solution variable u. The scheme also extends easily to third-order by imposing the exactness for cubic solutions. Note that the exactness is not a necessary condition for accuracy, but a sufficient condition to guarantee the design order of accuracy.

In the next two sections, we describe how to construct second- and third-order accurate schemes that, respectively, preserve quadratic and cubic solutions on arbitrary triangular grids.

5. Second-order scheme: RD-CC2

Consider the cell-residuals for the baseline hyperbolic RD scheme:

$$\Phi_u^E = \int_E (-au_x - bu_y + \nu p_x + \nu q_y + s)dxdy = \frac{1}{2}\sum_{i=1}^{3}\left(-a_{n_i}u_i|n_i| + \nu p_i n_{i_x} + \nu q_i n_{i_y}\right) + \overline{s}^E d\Omega^E, \quad (58)$$

$$\Phi_p^E = \frac{1}{T_r}\int_E (u_x - p)dxdy = \frac{1}{T_r}\left(\frac{1}{2}\sum_{i=1}^{3} u_i n_{i_x} - \overline{p}^E d\Omega^E\right), \quad (59)$$

$$\Phi_q^E = \frac{1}{T_r}\int_E (u_y - q)dxdy = \frac{1}{T_r}\left(\frac{1}{2}\sum_{i=1}^{3} u_i n_{i_y} - \overline{q}^E d\Omega^E\right). \quad (60)$$

These residuals are constructed based on the assumption that all variables (i.e., u, p, q) are linear over the element, and therefore they do not preserve quadratic solutions. Note that quadratic solutions imply linear gradients (i.e., u_x, u_y, p, q), and the above cell residuals are already exact for linear gradients. Therefore, to preserve quadratic solutions, the integrals of u_x and u_y terms need to be exact for quadratic solutions. A simple way to accomplish this is to reconstruct a quadratic element by interpolating the solution at the midpoint of each side in the element (see Fig. 4).

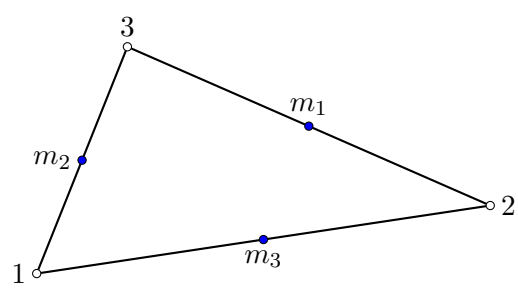


Figure 4: Reconstructed P_2 element with virtual midpoints.

The midpoint value u_{m_i} is estimated by the Hermite interpolation along the side:

$$u_{m_i} = \overline{u}_i - \frac{1}{8}\left(\Delta p_i \Delta x_i + \Delta q_i \Delta y_i\right), \quad (61)$$

where $\overline{u}_i$ is the arithmetic average of the solution values at the two end nodes, and $\Delta(\ )_i$ denotes the difference of the nodal values taken counterclockwise along the edge opposite to the node i, e.g., $\Delta x_3 = x_1 - x_2$. Note that the midpoint value u_{m_i} is exact for quadratic solutions (and linear gradients). Once the quadratic element is reconstructed, the cell-residuals are evaluated as line integrals with Simpson's rule applied along each side. This reconstruction approach was first introduced in Refs. [7, 31] and later employed in the form of the Green-Gauss gradient with a high-order curvature correction term [17, 8, 27]. In this work, we employ the latter implementation to upgrade the baseline residuals. Note that the formula (61) usually requires LSQ gradient reconstruction to obtain the nodal gradients [8, 7, 31, 27], but it is not necessary in the hyperbolic method because $p(= u_x)$ and $q(= u_y)$ are already available at nodes as a part of the primary variables. Note also that the edge midpiont values are introduced only artificially to derive the curvature correction terms and thus, do not appear in the final form of the cell residual.

Applying the curvature correction, we modify the cell-residuals of the baseline RD scheme as

$$\tilde{\Phi}_u^E = -\frac{1}{2}\sum_{i=1}^{3} a_{n_i}(u_i+\delta_i^u)|n_i| + \frac{\nu}{2}\sum_{i=1}^{3}\left(p_i n_{i_x} + q_i n_{i_y}\right) + \overline{s}^E d\Omega^E, \tag{62}$$

$$\tilde{\Phi}_p^E = \frac{1}{T_r}\left(\frac{1}{2}\sum_{i=1}^{3}(u_i+\delta_i^u)n_{i_x} - \overline{p}^E d\Omega^E\right), \tag{63}$$

$$\tilde{\Phi}_q^E = \frac{1}{T_r}\left(\frac{1}{2}\sum_{i=1}^{3}(u_i+\delta_i^u)n_{i_y} - \overline{q}^E d\Omega^E\right), \tag{64}$$

where the curvature correction term, δ_i^u, is defined as

$$\delta_i^u = \frac{1}{6}(\Delta p_i \Delta x_i + \Delta q_i \Delta y_i). \tag{65}$$

Note that in Ref. [17], δ_i^u was mistakenly shown with a negative sign. Observe also that these cell-residuals do not require the solution value at the edge midpoint. One may find it convenient that the scheme can be implemented as an add-on to the baseline hyperbolic RD scheme:

$$\tilde{\Phi}_u^E = \Phi_u^E - \frac{1}{2}\sum_{i=1}^{3} a_{n_i}\delta_i^u|n_i|, \tag{66}$$

$$\tilde{\Phi}_p^E = \Phi_p^E + \frac{1}{T_r}\left(\frac{1}{2}\sum_{i=1}^{3}\delta_i^u n_{x_i}\right), \tag{67}$$

$$\tilde{\Phi}_q^E = \Phi_q^E + \frac{1}{T_r}\left(\frac{1}{2}\sum_{i=1}^{3}\delta_i^u n_{y_i}\right). \tag{68}$$

The modified cell-residuals are distributed by the SUPG distribution matrix. The RD scheme based on these cell-residuals is referred to as the RD-CC2 scheme. This scheme is compact because no LSQ gradient reconstruction is necessary. Note that, we observed numerically that the linear solver diverges for high-Re cases unless LSQ gradient reconstruction is used in the evaluation of the curvature correction terms. The exactness for quadratic solutions has been confirmed numerically, and are compared with the baseline RD and RD-JCP2010 schemes. These results are summarized in Table 1,

Table 1: Exactness of the hyperbolic RD equations for exact quadratic and cubic polynomial solutions on irregular grids. The exact quadratic and cubic solutions are taken from Ref. [32]. The last column indicates whether the exactness (✓) is based on cell and/or nodal residuals.

Hyperbolic RD Scheme	Quadratic solution			Cubic solution			Cell/Nodal
	u eq.	p eq.	q eq.	u eq.	p eq.	q eq.	
Baseline	✗	✗	✗	✗	✗	✗	none
RD-JCP2010	✗	✓	✓	✗	✗	✗	cell
RD-CC2	✓	✓	✓	✗	✗	✗	both

The truncation error orders for the RD-CC2 scheme, the baseline RD scheme, and the RD-JCP2010 scheme are provided in Table 2. The truncation error orders have been determined by substituting a smooth exact solution (113) into the residuals for a series of irregular triangular grids. Clearly, all schemes have the same truncation error order for the nodal residual. A known theory for hyperbolic conservation laws [33, 34] states that the discretization error of $O(h^2)$ leads

to the truncation error of $O(h)$. However, our numerical results indicate that the converse is not necessarily true. As we will show later, only the RD-CC2 scheme, which has $O(h)$ truncation order, gives a discretization error of $O(h^2)$.

Table 2: Comparison between numerical truncation error of the nodal residuals (i.e., T.E. $= \frac{1}{d\Omega_j}\sum_j \mathbf{B}^E\mathbf{\Phi}^E$) obtained with RD-CC2 and non-genuine second-order RD schemes on irregular grids.

Hyperbolic RD Scheme	Nodal Residuals		
	u equation	p equation	q equation
Baseline	$O(h)$	$O(h)$	$O(h)$
RD-JCP2010	$O(h)$	$O(h)$	$O(h)$
RD-CC2	$O(h)$	$O(h)$	$O(h)$

We remark that the RD-CC2 scheme is different from the RD-JCP2010 scheme [17], which has the curvature correction only in $\tilde{\Phi}_p^E$ and $\tilde{\Phi}_q^E$ residuals. Thus, the whole residual vector of RD-JCP2010 is not exact for quadratic solutions as confirmed in Table 1. The RD-CC2 scheme, on the other hand, has the curvature correction applied to all the residuals, and therefore is exact for quadratic solutions (and linear gradients).

To solve the discrete equations, we employ the implicit solver described in Sec. 2.3. The RD-CC2 scheme is compact and therefore we linearize the residual vector exactly by using the AD tool. The implicit solver is thus Newton's method and converges to machine zero (i.e., ten orders of magnitude reduction in all the residuals) typically within 3–5 Newton iterations and 200–800 GS relaxations per Newtion iteration. Note that, the number of GS relaxations is the most relevant metric for the efficiency of the implicit solver, when the Jacobian is exact.

6. Third-order scheme: RD-CC3

To extend the compact RD-CC2 scheme to a third-order scheme, we modify the cell-residuals so that they preserve exact cubic solutions and quadratic gradients. Following the procedure explained in Sec. 5 for the RD-CC2, and thanks to the fact that both the Hermite interpolation and Simpson's rule are exact for cubic functions, we employ, again, the curvature correction technique. We emphasis again that similar to the construction of RD-CC2 scheme, we employ the curvature correction technique to achieve third-order accuracy and therefore the edge midpoint values are neither stored nor computed. Extension to the third-order, therefore, requires two additional steps: 1) high-order source term discretization, and 2) quadratic LSQ gradients for p, q, and the source term s. The high-order discretization of the source term is a critical piece for achieving third-order accuracy. We derive a formula by estimating the midpoint value by the Hermite interpolation and then exactly integrate the source terms assuming a quadratic variation over the element:

$$\int_E s\,dxdy = \left(\overline{s}^E - \frac{1}{4}\sum_{i=1}^{3}\delta_i^s\right) d\Omega^E, \tag{69}$$

$$\int_E p\,dxdy = \left(\overline{p}^E - \frac{1}{4}\sum_{i=1}^{3}\delta_i^p\right) d\Omega^E, \tag{70}$$

$$\int_E q\,dxdy = \left(\overline{q}^E - \frac{1}{4}\sum_{i=1}^{3}\delta_i^q\right) d\Omega^E, \tag{71}$$

where

$$\delta_i^s = \frac{1}{6}(\Delta s_{x_i}\Delta x_i + \Delta s_{y_i}\Delta y_i), \tag{72}$$

$$\delta_i^p = \frac{1}{6}(\Delta p_{x_i}\Delta x_i + \Delta p_{y_i}\Delta y_i), \tag{73}$$

$$\delta_i^q = \frac{1}{6}(\Delta q_{x_i}\Delta x_i + \Delta q_{y_i}\Delta y_i). \tag{74}$$

Using these high-order source discretizations, we construct the cell-residuals that preserve cubic solutions (and quadratic gradients) as follows:

$$\tilde{\tilde{\Phi}}_u^E = \Phi_u^E - \frac{1}{2}\sum_{i=1}^{3}\left(a_{n_i}\delta_i^u|n_i| - \nu(\delta_i^p n_{x_i} + \delta_i^q n_{y_i}) - \frac{1}{2}\delta_i^s d\Omega^E\right), \tag{75}$$

$$\tilde{\tilde{\Phi}}_p^E = \Phi_p^E + \frac{1}{2T_r}\sum_{i=1}^{3}\left(\delta_i^u n_{x_i} + \frac{1}{2}\delta_i^p d\Omega^E\right), \tag{76}$$

$$\tilde{\tilde{\Phi}}_q^E = \Phi_q^E + \frac{1}{2T_r}\sum_{i=1}^{3}\left(\delta_i^u n_{y_i} + \frac{1}{2}\delta_i^q d\Omega^E\right), \tag{77}$$

which can also be written in the form of an extension to the compact RD-CC2 scheme:

$$\tilde{\tilde{\Phi}}_u^E = \tilde{\Phi}_u^E + \frac{1}{2}\sum_{i=1}^{3}\left(\nu(\delta_i^p n_{x_i} + \delta_i^q n_{y_i}) + \frac{1}{2}\delta_i^s d\Omega^E\right), \tag{78}$$

$$\tilde{\tilde{\Phi}}_p^E = \tilde{\Phi}_p^E + \frac{1}{4T_r}\sum_{i=1}^{3}\delta_i^p d\Omega^E, \tag{79}$$

$$\tilde{\tilde{\Phi}}_q^E = \tilde{\Phi}_q^E + \frac{1}{4T_r}\sum_{i=1}^{3}\delta_i^q d\Omega^E. \tag{80}$$

The curvature correction terms require quadratic LSQ gradients for p, q, and s, but not for u because $p(= u_x)$ and $q(= u_y)$ are already available at nodes. We note that for high-Re cases, which will be discussed in more details in Sec. 8.1.2, LSQ reconstruction of the gradients is needed for the curvature correction terms that are applied to the first equation, otherwise the linear system diverges. This instability is observed for both linear and non-linear formulations of the hyperbolic advection-diffusion system. LSQ reconstruction does not affect the design order of accuracy of these schemes, as the quadratic and the cubic exactnesses are still preserved. Again, the cell-residuals are distributed by the SUPG distribution matrix. The hyperbolic RD scheme based on these cell-residuals is referred to as the RD-CC3 scheme. The exactness for cubic solutions has been confirmed numerically as shown in Table 3. The results of a numerical truncation error analysis for the RD-CC3 are provided in Table 4 and compared with the truncation error of RD-CC2.

Table 3: Exactness of the hyperbolic RD equations for exact quadratic and cubic polynomial solutions on irregular grids. The exact quadratic and cubic solutions are taken from Ref. [32]. The last column indicates whether the exactness (✓) is based on cell and/or nodal residuals.

Hyperbolic RD Scheme	Quadratic solution			Cubic solution			Elemental/Nodal
	u eq.	p eq.	q eq.	u eq.	p eq.	q eq.	
RD-CC2	✓	✓	✓	✗	✗	✗	both
RD-CC3	✓	✓	✓	✓	✓	✓	both

Table 4: Comparison between numerical truncation error of the nodal residuals (i.e., T.E. = $\frac{1}{d\Omega_j}\sum_j \mathbf{B}^E\mathbf{\Phi}^E$) obtained with the RD-CC2 and RD-CC3 schemes on irregular grids.

Hyperbolic RD Scheme	Nodal Residuals		
	u equation	p equation	q equation
RD-CC2	$O(h)$	$O(h)$	$O(h)$
RD-CC3	$O(h^2)$	$O(h^2)$	$O(h^2)$

The discrete equations are solved by the implicit solver described in Sec. 2.3. We do not attempt to linearize the RD-CC3 scheme exactly, but simply use the exact Jacobian of the compact RD-CC2 scheme. The implicit solver is therefore not precisely Newton's method, but it converges so rapidly that fully converged numerical solutions (i.e., ten orders of magnitude reductions for all residuals) are obtained typically within 10–15 Newton iterations and 200–800 Gauss-Seidel relaxations per Newton iteration as will be demonstrated later in Sec. 8.

Remark: The cell residuals evaluated with the RD-CC2 and the RD-CC3 schemes are conservative and have a telescoping property, because the curvature correction terms have been derived from Simpson's rule applied along each side of the reconstructed quadratic element (see Refs. [7, 35]). The line integral for the cells sharing the edge is exactly the same with an opposite sign and thus, we have conservation along each edge.

Also, the RD-CC2 and RD-CC3 schemes with the SUPG distribution matrix can be considered as economical and powerful alternatives to high-order finite-element methods [36]. They are economical mainly for three reasons: 1) the number of linear relaxations in the implicit solver increasers linearly, not quadratically as typical for diffusion problems, with grid refinement. 2) the second-order advection-diffusion scheme is compact, and thus allows an efficient construction of Newton's method. Furthermore, the same compact Jacobian still yields rapid convergence (comparable to Newton's method) for the third-order scheme. 3) second- and third-order accuracy can be achieved (as will be demonstrated later in Sec. 8) on linear triangular elements without introducing curved elements. Thus, high-order curved grids are not needed, and these schemes are directly applicable to existing grids composed of linear triangular elements. These schemes are also powerful in that they are capable of producing highly accurate and smooth gradients, to the same order of accuracy as that of the main solution variable u, on isotropic and anisotropic irregular grids. These advantages will be demonstrated in Sec. 8.

7. Nonlinear advection-diffusion equation

In this section, we describe the discretization of a nonlinear hyperbolic advection-diffusion equation. As an example, we discuss the construction of the RD-CC3 scheme with the SUPG distribution matrix. Note that the compact RD-CC2 is a subset of RD-CC3 scheme and therefore the discussion here also applies to the RD-CC2 scheme. A similar procedure can be applied to other distribution schemes. Throughout the discussion, we only consider the non-unified approach, which is applicable to more complex systems such as the compressible Navier-Stokes equations.

7.1. Nonlinear hyperbolic advection-diffusion equation

Consider the following two-dimensional nonlinear advection-diffusion equation:

$$\partial_t u + \partial_x f + \partial_y g = \partial_x(\nu\partial_x u) + \partial_y(\nu\partial_y u) + \tilde{s}(x, y, u), \tag{81}$$

where f and g are nonlinear functions of u, and $\nu = \nu(u)$. The advection speeds in x and y directions are therefore $a(u) = \partial f/\partial u$ and $b(u) = \partial g/\partial u$, respectively. Using the preconditioned

formulation proposed for nonlinear systems in Ref. [18], we construct a hyperbolic system as

$$\partial_\tau u + \partial_x f + \partial_y g = \partial_x p + \partial_y q + s, \tag{82}$$

$$\frac{T_r}{\nu}\partial_\tau p = \partial_x u - p/\nu, \tag{83}$$

$$\frac{T_r}{\nu}\partial_\tau q = \partial_y u - q/\nu, \tag{84}$$

where the variables p and q are, in the pseudo steady state, equivalent to the diffusive fluxes in x and y directions, respectively. As before, the physical time derivative can be incorporated into the source term s as is done in Ref. [24], but we focus on the steady equation here. Note that the system reduces to the target equation, i.e., Eq. (81), in the pseudo steady state. The system is written in the vector form as a preconditioned conservative equation, with the preconditioning matrix $\mathbf{P}$:

$$\mathbf{P}^{-1}\frac{\partial \mathbf{U}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \mathbf{Q}, \tag{85}$$

where

$$\mathbf{P}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & T_r/\nu(u) & 0 \\ 0 & 0 & T_r/\nu(u) \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} s \\ -p/\nu(u) \\ -q/\nu(u) \end{bmatrix}, \tag{86}$$

$$\mathbf{F} = \mathbf{F}^a + \mathbf{F}^d = \begin{bmatrix} f \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -p \\ -u \\ 0 \end{bmatrix}, \quad \mathbf{G} = \mathbf{G}^a + \mathbf{G}^d = \begin{bmatrix} g \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -q \\ 0 \\ -u \end{bmatrix}. \tag{87}$$

Note that the presence of the preconditioning matrix $\mathbf{P}$ is mainly to simplify the construction of the numerical scheme, and not to improve the condition number of the differential system (see Ref. [18]).

The flux Jacobians can be obtained by multiplying both sides of Eq. (85) by the preconditioned matrix $\mathbf{P}$, and arrive at

$$\mathbf{P}\mathbf{F}_x = \mathbf{P}\frac{\partial \mathbf{F}}{\partial \mathbf{U}}\mathbf{U}_x = \mathbf{A}\mathbf{U}_x, \quad \mathbf{P}\mathbf{G}_y = \mathbf{P}\frac{\partial \mathbf{G}}{\partial \mathbf{U}}\mathbf{U}_y = \mathbf{B}\mathbf{U}_y, \tag{88}$$

where

$$\mathbf{A} = \begin{bmatrix} a(u) & -1 & 0 \\ -\nu(u)/T_r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b(u) & 0 & -1 \\ 0 & 0 & 0 \\ -\nu(u)/T_r & 0 & 0 \end{bmatrix}. \tag{89}$$

Note that these preconditioned Jacobian matrices (i.e., $\mathbf{A}$ and $\mathbf{B}$) are slightly different from the ones obtained for the linear hyperbolic advection-diffusion system. Hence, their eigen-structures are also different from those given for the linear system in Sec. 2. The differences in the eigen-structures influence the formulation of the distribution matrices as described next. Below, we first describe the baseline RD scheme for the preconditioned nonlinear system, and then upgrade it to a higher order by the correction approach.

7.2. Baseline RD scheme

The cell-residual for the preconditioned system is evaluated in two steps. First, we evaluate the cell-residuals of the nonlinear equation by integrating the right hand side of Eq. (85) over an element $E \in \{E\}$ to get the unpreconditioned cell-residuals $\mathbf{\Psi}$:

$$\begin{aligned} \mathbf{\Psi}^E = \begin{bmatrix} \Psi_u^E \\ \Psi_p^E \\ \Psi_q^E \end{bmatrix} &= \int_E (-\mathbf{F}_x - \mathbf{G}_y + \mathbf{Q}) dxdy, \\ &= -\frac{1}{2}\sum_{i=1}^{3}\left(\mathbf{F}_i \hat{n}_{i_x} + \mathbf{G}_i \hat{n}_{i_y}\right)|\mathbf{n}_i| + \overline{\mathbf{Q}}^E d\Omega^E, \end{aligned} \tag{90}$$

The preconditioned cell-residual, which is distributed to the element vertices, is then defined as

$$\mathbf{\Phi}^E = \begin{bmatrix} \Phi_u^E \\ \Phi_p^E \\ \Phi_q^E \end{bmatrix} = \overline{\mathbf{P}}\mathbf{\Psi}^E \tag{91}$$

where $\overline{\mathbf{P}}$ is evaluated by the arithmetic average of the solution $\mathbf{U}$ within the element. The matrix $\mathbf{K}_i$ corresponding to the preconditioned system is defined by

$$\begin{aligned} \mathbf{K}_i &= \frac{1}{2}\overline{\mathbf{P}}\left(\frac{\partial \mathbf{F}}{\partial \mathbf{U}}\hat{n}_{i_x} + \frac{\partial \mathbf{G}}{\partial \mathbf{U}}\hat{n}_{i_y}\right)|\mathbf{n}_i|, \\ &= \mathbf{K}_i^a + \mathbf{K}_i^d, \end{aligned} \tag{92}$$

where $\mathbf{K}_i^a$ and $\mathbf{K}_i^d$ are, respectively, the contributions from the advection and diffusion components, defined as

$$\mathbf{K}_i^a = \frac{1}{2}\overline{\mathbf{P}}\left(\frac{\partial \mathbf{F}^a}{\partial \mathbf{U}}n_{i_x} + \frac{\partial \mathbf{G}^a}{\partial \mathbf{U}}n_{i_y}\right) = \frac{1}{2}\mathbf{A}_{n_i}^a|\mathbf{n}_i|, \tag{93}$$

$$\mathbf{K}_i^d = \frac{1}{2}\overline{\mathbf{P}}\left(\frac{\partial \mathbf{F}^d}{\partial \mathbf{U}}n_{i_x} + \frac{\partial \mathbf{G}^d}{\partial \mathbf{U}}n_{i_y}\right) = \frac{1}{2}\mathbf{A}_{n_i}^d|\mathbf{n}_i|, \tag{94}$$

where

$$\mathbf{A}_{n_i}^a = \begin{bmatrix} a_{n_i} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{A}_{n_i}^d = \begin{bmatrix} 0 & -\hat{n}_{i_x} & -\hat{n}_{i_y} \\ -\overline{\nu}\hat{n}_{i_x}/T_r & 0 & 0 \\ -\overline{\nu}\hat{n}_{i_y}/T_r & 0 & 0 \end{bmatrix}, \tag{95}$$

and $\overline{\nu}$ is the arithmetic average of the diffusion coefficient over the element, and

$$a_n = \frac{\partial f}{\partial u}\hat{n}_x + \frac{\partial g}{\partial u}\hat{n}_y. \tag{96}$$

It is clear that the eigenvalues of the advective and diffusive components of the nonlinear hyperbolic advection-diffusion system are equivalent to the linear system:

$$\lambda^a = a_n, \quad \lambda_1^d = -\sqrt{\frac{\overline{\nu}}{T_r}}, \quad \lambda_2^d = +\sqrt{\frac{\overline{\nu}}{T_r}}. \tag{97}$$

Similarly to the linear advection-diffusion system, we construct the SUPG distribution matrix of the nonlinear system over the element E as:

$$\mathbf{B}_i^{\text{SUPG}} = \frac{1}{3}\mathbf{I} + \mathbf{D}_i^a + \mathbf{D}_i^d, \tag{98}$$

where

$$\mathbf{D}_i^a = \begin{bmatrix} d_i^{\text{SUPG}} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad d_i^{\text{SUPG}} = \frac{1}{2}\frac{\overline{a}_{n_i}|n_i|}{\sum_{l=1}^3 \max(0, \overline{a}_{n_l})|n_l|}, \tag{99}$$

and

$$\mathbf{D}_i^d = \frac{1}{2}\frac{\mathbf{K}_i^d}{\sum_{l=1}^3 \mathbf{K}_l^{d+}}, \quad \mathbf{K}_i^d = \frac{1}{2}\mathbf{A}_{n_i}^d|\mathbf{n}_i|, \tag{100}$$

where $\mathbf{K}_i^{d+}$ is constructed based on the projection of the $\mathbf{A}_n^d$ onto the λ_2^d running wave. We note here that the diffusive flux Jacobian matrix $\mathbf{A}_n^d$ has the following eigenvectors:

$$\mathbf{R}_n = \begin{bmatrix} -1/\lambda_1 & -1/\lambda_2 & 0 \\ \hat{n}_x & \hat{n}_x & -\hat{n}_y \\ \hat{n}_y & \hat{n}_y & \hat{n}_x \end{bmatrix}, \quad \mathbf{L}_n = \frac{1}{\lambda_1 - \lambda_2}\begin{bmatrix} \lambda_1\lambda_2 & \lambda_1\hat{n}_x & \lambda_1\hat{n}_y \\ -\lambda_1\lambda_2 & -\lambda_2\hat{n}_x & -\lambda_2\hat{n}_y \\ 0 & -(\lambda_1 - \lambda_2)\hat{n}_y & (\lambda_1 - \lambda_2)\hat{n}_x \end{bmatrix}, \tag{101}$$

which is different from the linear case. Thus, the projection matrix $\mathbf{\Pi}_2$ for the nonlinear case becomes

$$\mathbf{\Pi}_2 = \frac{1}{\lambda_1 - \lambda_2} \begin{bmatrix} \lambda_1 & \hat{n}_x & \hat{n}_y \\ -\lambda_1\lambda_2\hat{n}_x & -\lambda_2\hat{n}_x^2 & -\lambda_2\hat{n}_x\hat{n}_y \\ -\lambda_1\lambda_2\hat{n}_y & -\lambda_2\hat{n}_x\hat{n}_y & -\lambda_2\hat{n}_y^2 \end{bmatrix}, \tag{102}$$

and the $\mathbf{K}_i^{d+}$ can be expressed as

$$\mathbf{K}_i^{d+} = \frac{1}{2}\lambda_{2,i}^d \mathbf{\Pi}_{2,i}|\mathbf{n}_i|. \tag{103}$$

The residual at a node j is then defined in the form of Eq. (10). This completes the construction of the baseline RD scheme for the nonlinear advection-diffusion equation.

7.3. RD-CC2 and RD-CC3 schemes

The improved schemes, RD-CC2 and RD-CC3, can be constructed by improving the cell-residuals with the high-order curvature correction approach discussed in Secs. 5 and 6. Special attention should be paid, however, when constructing the high-order hyperbolic RD schemes for nonlinear equations: the Δp and Δq terms in Eq. (65) require the solution gradients, while p and q variables in the nonlinear formulation are the diffusive fluxes. With this in mind, the construction is straightforward.

The cell-residuals of the RD-CC3 scheme for the nonlinear advection-diffusion equation are given by

$$\tilde{\tilde{\Phi}}_u^E = \Phi_u^E - \frac{1}{2}\sum_{i=1}^{3}\left[\left(\delta_i^f - \delta_i^p\right) n_{x_i} + \left(\delta_i^g - \delta_i^q\right) n_{y_i} + \frac{1}{2}\delta_i^s d\Omega^E\right], \tag{104}$$

$$\tilde{\tilde{\Phi}}_p^E = \Phi_p^E + \frac{\overline{\nu}}{2T_r}\sum_{i=1}^{3}\left(\delta_i^u n_{x_i} + \frac{1}{2}\delta_i^{p/\nu} d\Omega^E\right), \tag{105}$$

$$\tilde{\tilde{\Phi}}_q^E = \Phi_q^E + \frac{\overline{\nu}}{2T_r}\sum_{i=1}^{3}\left(\delta_i^u n_{y_i} + \frac{1}{2}\delta_i^{q/\nu} d\Omega^E\right), \tag{106}$$

where δ_i^s, δ_i^p and δ_i^q are given by Eqs. (72), (73), and (74), respectively, and

$$\delta_i^f = \frac{1}{6}\left(\Delta f_{x_i}\Delta x_i + \Delta f_{y_i}\Delta y_i\right), \tag{107}$$

$$\delta_i^g = \frac{1}{6}\left(\Delta g_{x_i}\Delta x_i + \Delta g_{y_i}\Delta y_i\right), \tag{108}$$

$$\delta_i^u = \frac{1}{6}\left(\Delta(p/\nu)_i\Delta x_i + \Delta(q/\nu)_i\Delta y_i\right), \tag{109}$$

$$\delta_i^{p/\nu} = \frac{1}{6}\left(\Delta(p/\nu)_{x_i}\Delta x_i + \Delta(p/\nu)_{y_i}\Delta y_i\right), \tag{110}$$

$$\delta_i^{q/\nu} = \frac{1}{6}\left(\Delta(q/\nu)_{x_i}\Delta x_i + \Delta(q/\nu)_{y_i}\Delta y_i\right). \tag{111}$$

For the flux gradients in the curvature correction terms for f and g, we employ the chain rule to avoid the LSQ gradient reconstruction:

$$f_x = \frac{\partial f}{\partial u}u_x = \frac{\partial f}{\partial u}(p/\nu), \quad f_y = \frac{\partial f}{\partial u}u_y = \frac{\partial f}{\partial u}(q/\nu), \tag{112}$$

and similarly for the flux g. The gradients of s, p, q, p/ν, q/ν, and the source term, s, are computed by the quadratic LSQ fit. The RD-CC3 scheme is obtained by distributing the above residuals by the SUPG distribution matrix. The RD-CC2 scheme is obtained simply by removing δ_i^p, δ_i^q, δ_i^s, $\delta_i^{p/\nu}$, and $\delta_i^{q/\nu}$ terms from the cell-residuals. Consequently, the RD-CC2 scheme does not require any LSQ gradient reconstruction and therefore, is compact as in the linear case. We confirmed by numerical experiments that the RD-CC2 and RD-CC3 schemes preserve exact quadratic and cubic solutions, respectively, by the method of manufactured solutions.

8. Results

In this section, we verify and examine the accuracy of the hyperbolic RD schemes. In all the test problems, we specify the exact solution u on the boundaries, and initialize all the variables in the interior of the domain with a zero value, and solve the hyperbolic system for u, p, and q. The linear relaxation is performed with a Gauss-Seidel (GS) algorithm to reduce the linear residual by five orders of magnitude with the maximum of 1000 relaxations. Typically, it takes 300–600 relaxations with the under-relaxation parameter in the range of 0.5 to 0.8. The implicit solver is taken to be converged at ten orders of magnitude residual reduction for all equations; this is obtained typically with 3–5 Newton iterations for the RD-CC2 scheme, and 10–15 Newton iterations for the RD-CC3 scheme.

Simulations were performed using both the *unified* and the *non-unified* approaches discussed in Sec. 3. The convergence speed and the order of accuracy results were found to be nearly identical with either of the two approaches. Therefore, we only present the results obtained with the *non-unified* approach.

8.1. Linear advection-diffusion problem

For the linear advection-diffusion equation (1), we verify the order of accuracy and the quality of the predicted solution and solution gradients for isotropic and anisotropic irregular triangular grids, and for a problem with curved boundaries.

8.1.1. Isotropic Grids

Consider a steady two-dimensional linear advection-diffusion equation, Eq. (1), with $\tilde{s} = 0$. This equation has an exact exponential solution [9] of the form:

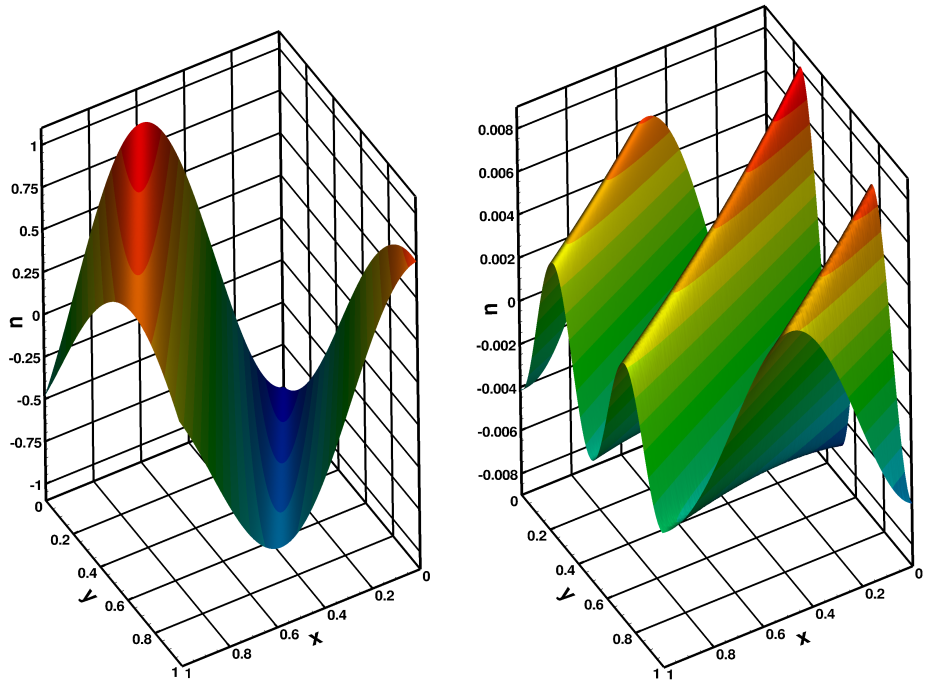
$$u(x, y) = C \cos(A\pi\eta) \exp\left(\frac{1 - \sqrt{1 + 4A^2\pi^2\nu^2}}{2\nu}\xi\right), \tag{113}$$

where A and C are arbitrary constants, and $\xi = ax + by$ and $\eta = bx - ay$. This solution is smooth and has no singularly on the boundaries (see Fig. 5) and therefore, is appropriate for the order of accuracy verification.

A series of isotropic irregular grids was generated and the accuracy of the proposed high-order schemes is verified against the baseline RD and RD-JCP2010 schemes. The first three coarse grids are shown in Fig. 6. Error convergence results are shown in Fig. 7 along with more detailed information in Tables 5–7. The error convergence was obtained by computing the L_1 norms defined as $\mathbf{L}_1(:) = \sum_{j=1}^{N} |\mathbf{U}_j - \mathbf{U}_j^e|/N$, where $\mathbf{U}$ is the solution vector, the superscript e denotes the exact value, and N is the total number of nodes (Note: the same order of accuracy results are obtained with $\mathbf{L}_2$ error, and thus not shown). For the variable u, all second-order schemes give truly second-order accuracy, and the RD-CC3 scheme gives fourth-order accuracy. For the gradient variables, p and q, on the other hand, the RD-CC2 and RD-CC3 schemes yield, respectively, second- and third-order accuracy, but the baseline RD and RD-JCP2010 do not give truly second-order accuracy.

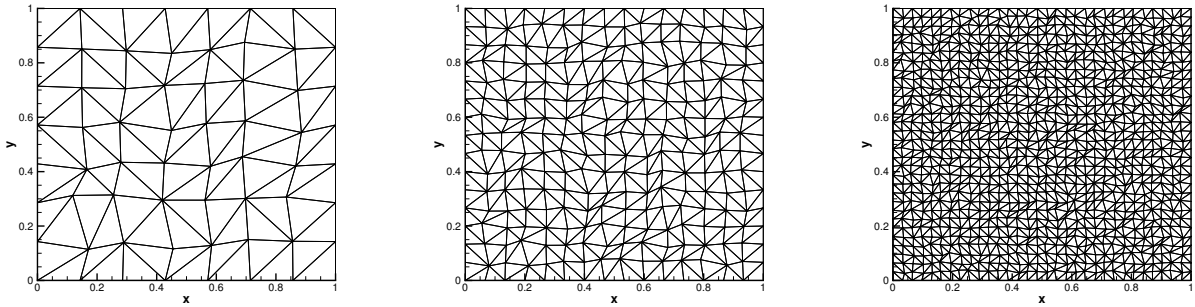
Table 5: $L1$ error convergence of u for hyperbolic RD schemes with the SUPG distribution on irregular grids ($A = 2$, $C = -1$, $a = 2$, $b = 1$, $\nu = 0.01$).

Grids	RD	Order	RD-JCP2010	Order	RD-CC2	Order	RD-CC3	Order
32×32	5.66E-05	–	6.98E-05	–	1.91E-05	–	2.31E-06	–
64×64	1.08E-05	2.39	1.31E-05	2.41	4.11E-06	2.22	1.43E-07	4.01
128×128	2.47E-06	2.13	2.82E-06	2.22	9.87E-07	2.06	1.02E-08	3.81
256×256	5.73E-07	2.11	6.18E-07	2.19	2.38E-07	2.02	6.94E-10	3.88



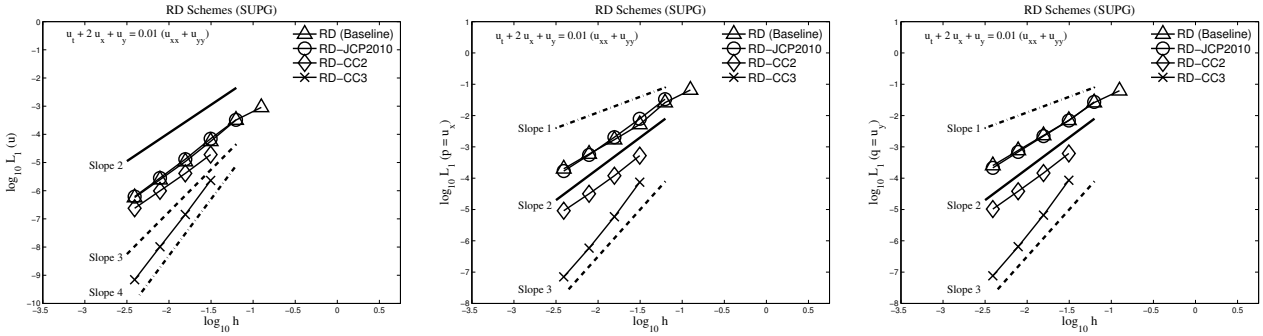
(a) $A = 2$, $C = -1$, $a = 0.50$, $b = 0.85$ (b) $A = 2$, $C = -0.009$, $a = 2.0$, $b = 1.0$

Figure 5: Graphical representation of two exact solutions ($\nu = 0.01$).



(a) 8×8 (b) 16×16 (c) 32×32

Figure 6: Samples of the irregular grids used in this study.



(a) u (b) $p = u_x$ (c) $q = u_y$

Figure 7: Order of accuracy comparisons of the baseline RD, RD-JCP2010, RD-CC2, and RD-CC3 schemes.

Table 6: $L1$ error convergence of p $(= u_x)$ for hyperbolic RD schemes with the SUPG distribution on irregular grids ($A = 2$, $C = -1$, $a = 2$, $b = 1$, $\nu = 0.01$).

Grids	RD	Order	RD-JCP2010	Order	RD-CC2	Order	RD-CC3	Order
32×32	5.19E-03	–	7.87E-03	–	5.25E-04	–	7.43E-05	–
64×64	1.70E-03	1.61	2.04E-03	1.94	1.20E-04	2.13	5.88E-06	3.66
128×128	5.88E-04	1.53	5.54E-04	1.88	3.18E-05	1.92	5.85E-07	3.33
256×256	1.99E-04	1.56	1.69E-04	1.71	9.14E-06	1.80	7.06E-08	3.05

Table 7: $L1$ error convergence of q $(= u_y)$ for hyperbolic RD schemes with the SUPG distribution on irregular grids ($A = 2$, $C = -1$, $a = 2$, $b = 1$, $\nu = 0.01$).

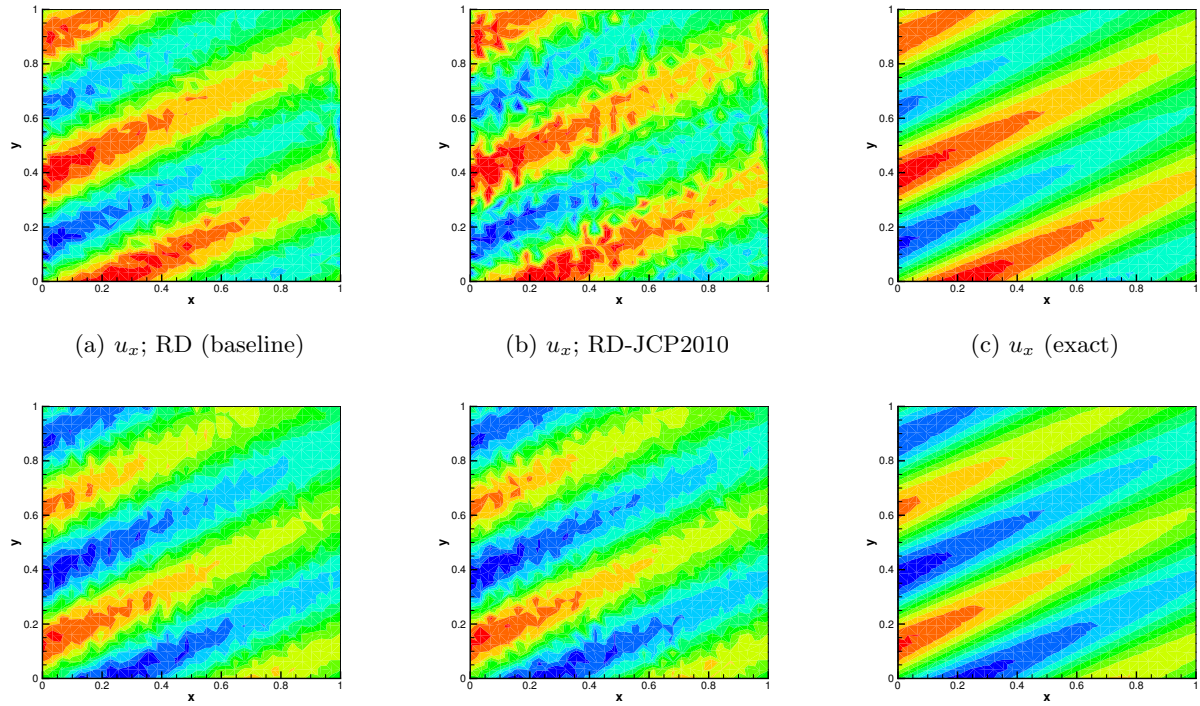
Grids	RD	Order	RD-JCP2010	Order	RD-CC2	Order	RD-CC3	Order
32×32	7.04E-03	–	6.93E-03	–	6.16E-04	–	8.58E-05	–
64×64	2.32E-03	1.60	2.20E-03	1.65	1.46E-04	2.08	6.61E-06	3.70
128×128	7.67E-04	1.60	6.91E-04	1.67	3.82E-05	1.93	6.50E-07	3.35
256×256	2.54E-04	1.59	2.14E-04	1.69	1.03E-05	1.89	7.62E-08	3.09

Table 8: Iterative convergence of the various hyperbolic RD schemes on irregular grids ($A = 2$, $C = -1$, $a = 2$, $b = 1$, $\nu = 0.01$). Solutions are considered converged when the minimum of ten orders of magnitude reduction is achieved for all the nodal-residuals. The Gauss-Seidel relaxation is performed for each Newton iteration until at least five orders of magnitude reduction is obtained in the linear residuals.

Grids	RD		RD-JCP2010		RD-CC2		RD-CC3	
	Newton (Nt)	GS/Nt	Newton	GS/Nt	Newton	GS/Nt	Newton	GS/Nt
32×32	3	68	3	48	3	72	10	76
64×64	3	103	3	69	3	76	10	83
128×128	3	165	3	111	3	118	10	170
256×256	3	273	3	190	3	249	10	361

Iterative convergence results are provided in Table 8 with the total number of Newton iterations and the number of GS linear relaxations per Newton iteration. As expected, the implicit solver, which is Newton's method for the baseline, the RD-JCP2010, and the RD-CC2 schemes, converges with 3–5 Newton iterations with an average of about 500 GS relaxations per Newton iteration. For the RD-CC3 scheme, the implicit solver, which is not truly Newton's method, converges also very rapidly in 10 Newton iterations. Note also that asymptotically the number of linear relaxations increases linearly (not quadratically as is typical in conventional methods) with the grid sizes. This is a feature of the hyperbolic method, arising from the elimination of the second derivatives in the target equation [16].

To demonstrate the high quality of the predicted solution gradients within the domain by the improved RD-CC2 and RD-CC3 schemes, we show results for the 32×32 irregular grid shown in Fig. 6b. Accurate prediction of the solution gradients within the domain is of great interest in many applications (e.g., turbulent flows, reacting flows, etc.). Figure 8 shows the solution gradients predicted by the baseline RD and RD-JCP2010 schemes, and compares them with the exact solutions. As can be seen clearly, the predicted results are very oscillatory and inaccurate.



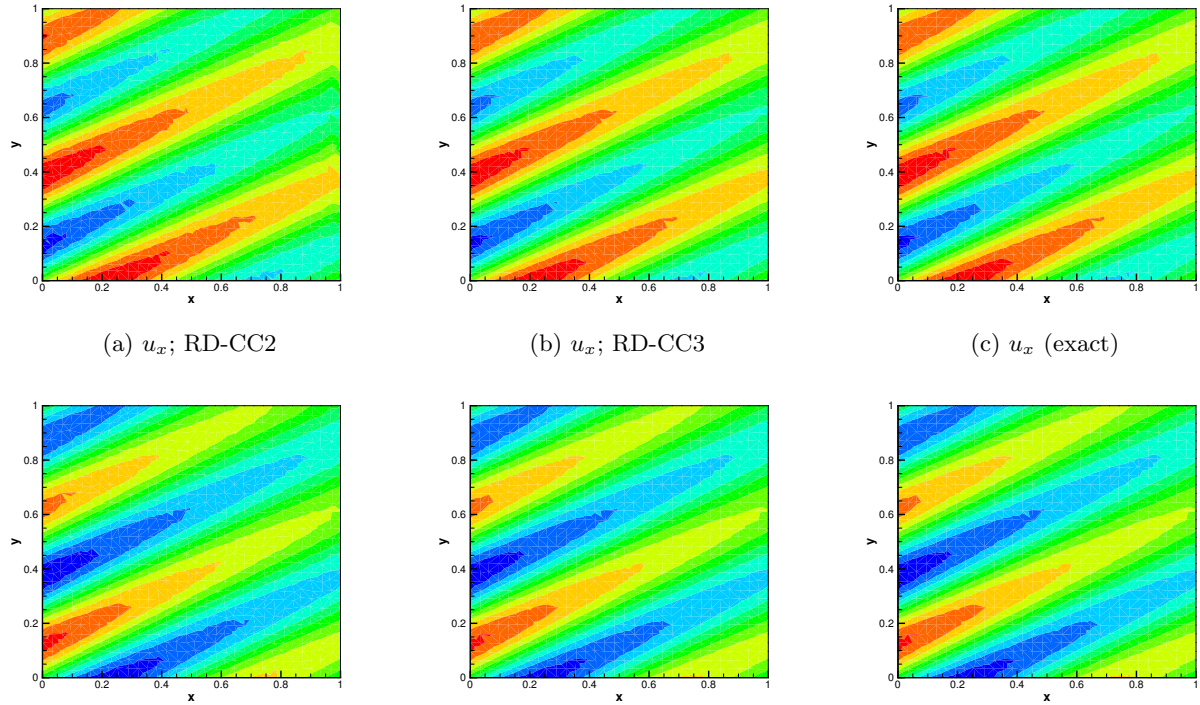
(a) u_x; RD (baseline) (b) u_x; RD-JCP2010 (c) u_x (exact)

(d) u_y; RD (baseline) (e) u_y; RD-JCP2010 (f) u_y (exact)

Figure 8: Comparisons between solution gradients (u_x and u_y) predicted by the baseline and RD-JCP2010 schemes on irregular 32×32 grid (see Fig. 6b) with the exact values ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $\nu = 0.01$).

On the other hand, as shown in Fig. 9, the gradients predicted by the proposed RD-CC2 and RD-CC3 schemes are very accurate and smooth. Note that, our numerical experiments suggest that in some cases the noise produced by the baseline RD and RD-JCP2010 schemes will not completely disappear in the entire domain with further grid refinement.

The reconstructed gradients computed by the quadratic LSQ fits from the solution of the RD-CC3 scheme, compared with the gradients predicted directly with the RD-CC3 scheme are shown in Figs. 10 and 11. Clearly the reconstructed gradients are less accurate and deteriorate even more within the domain. This comparison emphasizes the difficulty in obtaining accurate solution



(a) u_x; RD-CC2 (b) u_x; RD-CC3 (c) u_x (exact)

(d) u_y; RD-CC2 (e) u_y; RD-CC3 (f) u_y (exact)

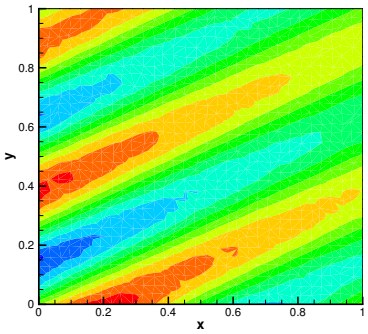
Figure 9: Comparisons between solution gradients (u_x and u_y) predicted by the RD-CC2 and RD-CC3 schemes on irregular 32×32 grid (see Fig. 6b) with the exact values ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $\nu = 0.01$).

gradients by reconstruction techniques, a common practice with conventional schemes.

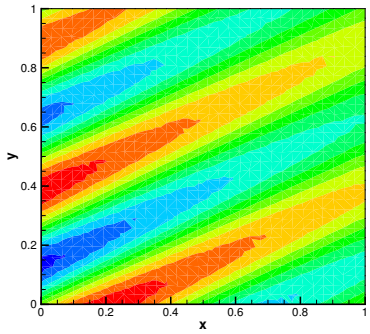
8.1.2. Uniform accuracy over $Re = a/\nu$

In this section we demonstrate uniform accuracy predicted by the RD-CC2 and RD-CC3 schemes for both the solution and the solution gradients over a range of Re with different irregular grids. These results are shown in Fig. 12, which verifies that the RD-CC2 and RD-CC3 schemes produce, respectively, at a minimum, second- and third-order solution for all the variables. Note that in the advection-limit ($\nu \to 0$) the hyperbolic advection-diffusion system reduces to a simple scalar advection equation, for which the RD-CC2 scheme is third-order accurate for all the variables. In the diffusion-limit ($\nu \to \infty$), the hyperbolic advection-diffusion system becomes hyperbolic diffusion system, for which the RD-CC3 schemes is fourth-order for u but remains third-order for the solution gradients.

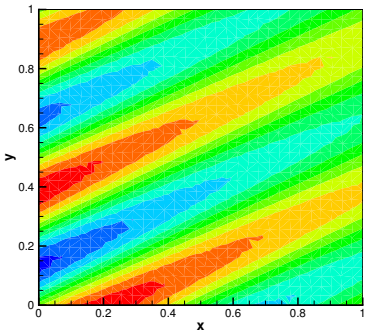
Figure 13 shows the Newton iteration versus Re for both RD-CC2 and RD-CC3 schemes. It is clear that the number of Newton iteration is independent of the Re and grid sizes. We remark that some instabilities were observed with the linear relaxation (i.e., the relaxation diverges) for high-Re cases ($Re \geq 10^3$), but found that both RD-CC2 and RD-CC3 schemes become stable if the curvature correction term applied to the advection fluxes (i.e., δ_i^u), is evaluated with LSQ reconstructions; that is, u_x and u_y variables present in the δ_i^u term (see Eq. 65) are reconstructed from the solution variable u, instead of being substituted by the gradient variables p and q. The jump in the number of Newton iterations shown in Fig. 13 for RD-CC2 is due to this switch, which is implemented to avoid instability for high-Re cases. As it is shown, the design order of accuracy of these schemes is obtained for all, including the high-Re, cases. We also remark that if the switch is activated for all the Re, the jump will disappear and a similar result shown for RD-CC3 scheme will be recovered. Note that the technique proposed for the high-Re cases can be directly applied



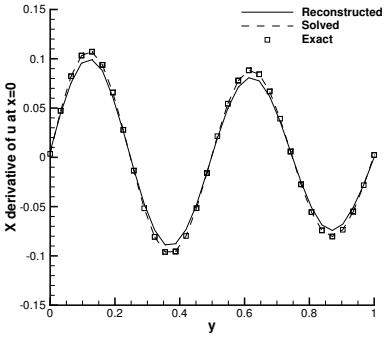
(a) Reconstructed; u_x



(b) Predicted; u_x

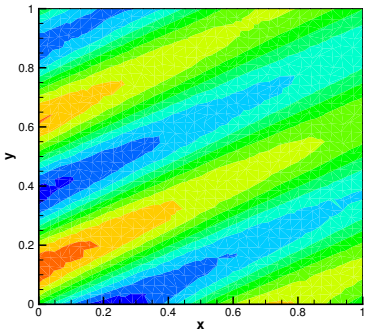


(c) Exact; u_x

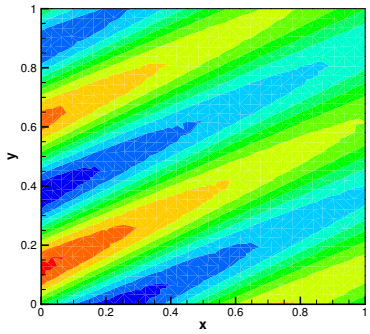


(d) u_x at $x = 0$

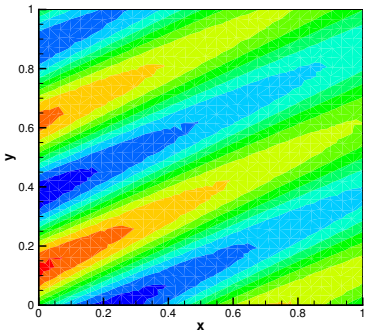
Figure 10: Comparison of u_x reconstructed with the quadratic LSQ fit from the solution of the RD-CC3 scheme and the one predicted directly with the RD-CC3 scheme on irregular 32×32 grid in Fig. 6b. ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $\nu = 0.01$).



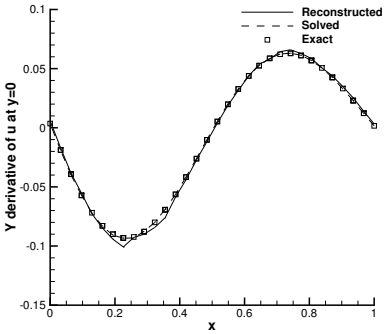
(a) Reconstructed; u_y



(b) Predicted; u_y

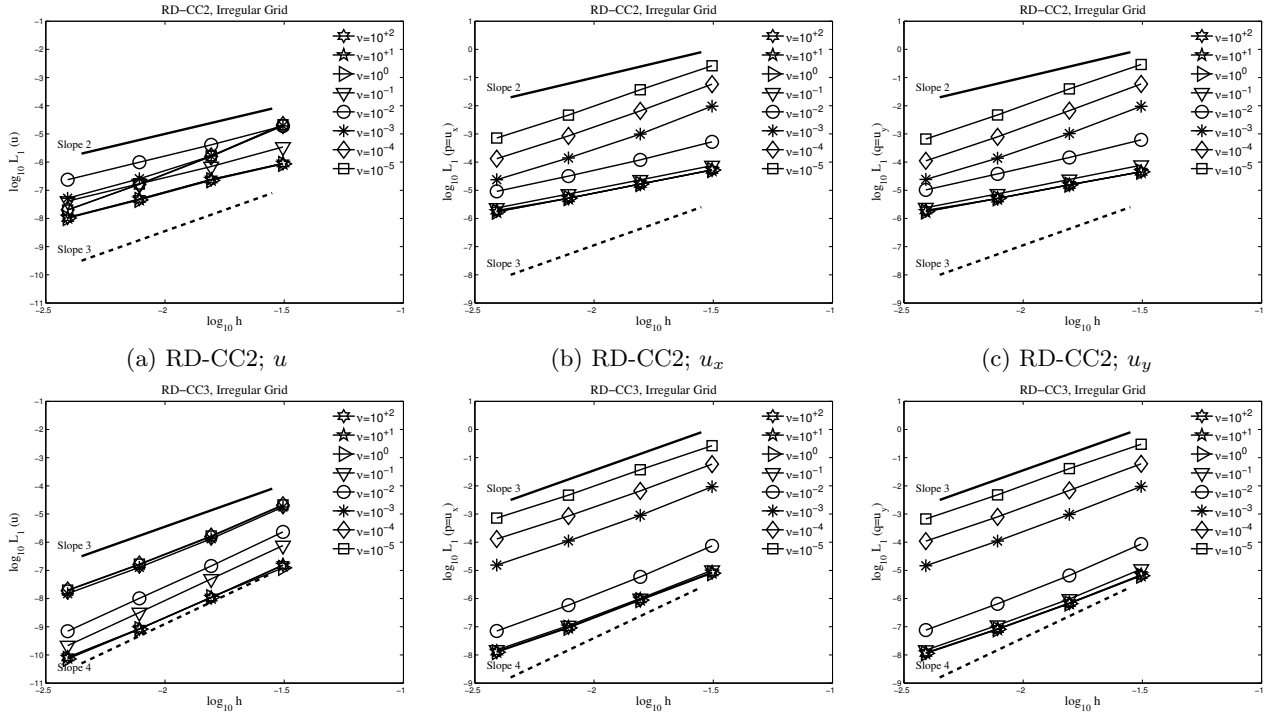


(c) Exact; u_y



(d) u_y at $y = 0$

Figure 11: Comparison of u_y reconstructed with the quadratic LSQ fit from the solution of the RD-CC3 scheme and the one predicted directly with the RD-CC3 scheme on irregular 32×32 grid in Fig. 6b. ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $\nu = 0.01$).

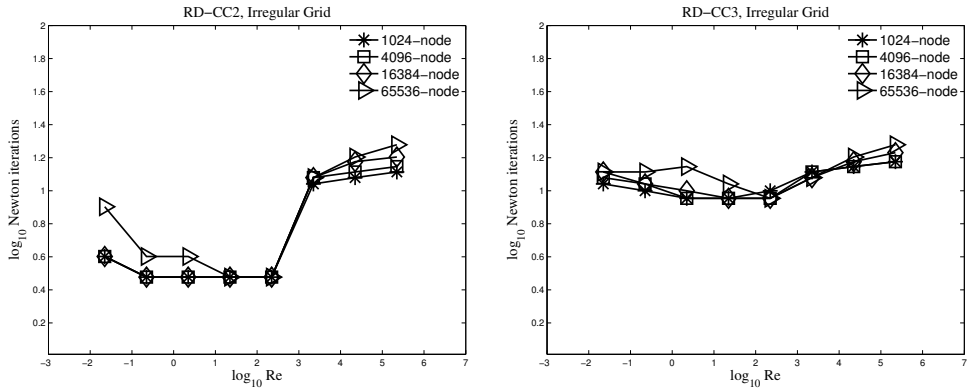


(a) RD-CC2; u (b) RD-CC2; u_x (c) RD-CC2; u_y

(d) RD-CC3; u (e) RD-CC3; u_x (f) RD-CC3; u_y

Figure 12: Uniform accuracy over a range of Re with the proposed RD-CC2 and RD-CC3 schemes on irregular grids ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $Re = \sqrt{5}/\nu$).

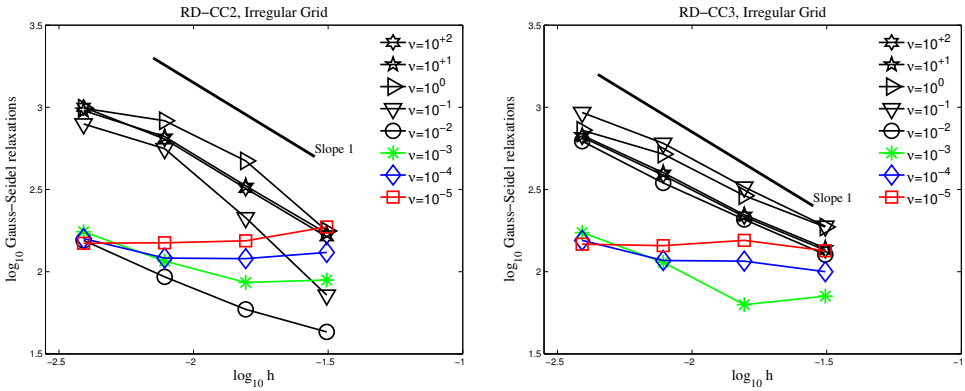
to practical applications in solving the Navier-Stokes equation, as the curvature correction terms are computed separately for each equation, and added to the baseline residuals.



(a) RD-CC2 (b) RD-CC3

Figure 13: Newton iterations used by the proposed RD-CC2 and RD-CC3 schemes on irregular grids over a range of Re and grids ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $Re = \sqrt{5}/\nu$).

A linear convergence, $O(N)$, was also obtained for the range of Re. This is shown in Fig. 14. For high-Re cases, a better than linear convergence is obtained because of high mesh-Re (i.e., ah/ν) values used with coarser grids. For example, for the $\nu = 10^{-3}$ case ($Re = 1000\sqrt{5}$), linear convergence is obtained from the second coarsest grid (mesh-$Re < 35$). Note that the RD-CC2 and RD-CC3 schemes are efficient enough in converging high-Re cases on grids with high mesh-Re, and therefore we did not attempt to fix a mesh-Re on these cases.



(a) RD-CC2 (b) RD-CC3

Figure 14: Average Gauss-Seidel relaxations (100–300) per Newton iteration used by the proposed RD-CC2 and RD-CC3 schemes on irregular grids over a range of Re and grids ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $Re = \sqrt{5}/\nu$).

8.1.3. High aspect-ratio grids

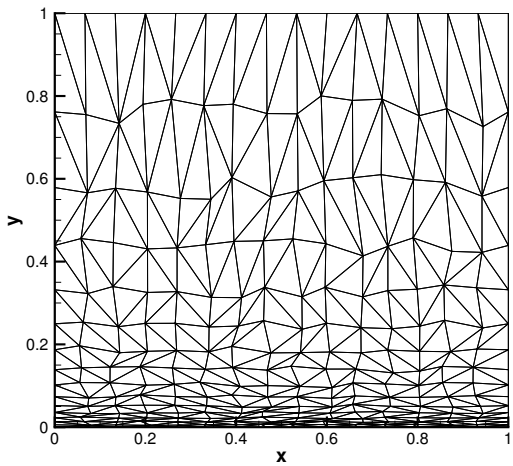
In practical applications, we are often interested in the gradients at the boundary and with highly anisotropic grids, e.g., viscous drag or heat flux predictions in high-Reynolds-number viscous flows. To demonstrate the benefit of the hyperbolic schemes for such applications, we consider two highly stretched anisotropic irregular grids (see Fig. 15). The y-derivative of u at $y = 0$ is plotted and compared with the exact solution in Fig. 15. The predicted normal gradient is very accurate and smooth on such relatively coarse irregular grids. Similar results have been obtained by other schemes, and therefore are not shown.

This exceptional feature of the hyperbolic RD schemes is further illustrated by comparing the normal gradients predicted by the baseline RD scheme and a conventional FV scheme on a high aspect-ratio (AR) grid. The grid and the results are shown in Fig. 16 where the normal gradients at the $y = 0$ and the $x = 0$ boundaries predicted with the hyperbolic RD scheme are compared with those obtained by the linear LSQ reconstruction from the solution of a conventional FV scheme taken from Ref. [19], where the scalar advection-diffusion equation is solved by the second-order edge–based advection scheme and the second-order Galerkin discretization for diffusion (referred to as Galerkin in the figure). The predicted normal derivative using the baseline RD scheme is remarkably smooth and accurate, while the gradients obtained by the conventional scheme shows oscillatory behavior and inaccurate predictions. For this problem also, similar results have been obtained by other schemes, and therefore are not shown.

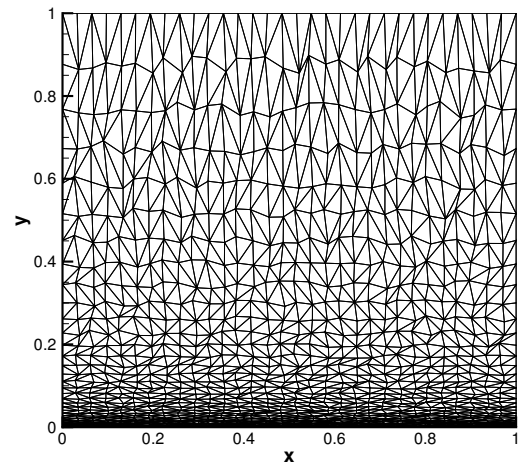
These results indicate that even the baseline RD and RD-JCP2010 schemes can produce accurate gradients on boundaries although they yield oscillations in the interior as shown in the previous section. Also, the results demonstrate that the RD-CC2 and RD-CC3 schemes perform successfully even on irregular high-aspect-ratio grids, which are representatives of adapted viscous grids.

8.2. Domain with curved geometrical boundaries

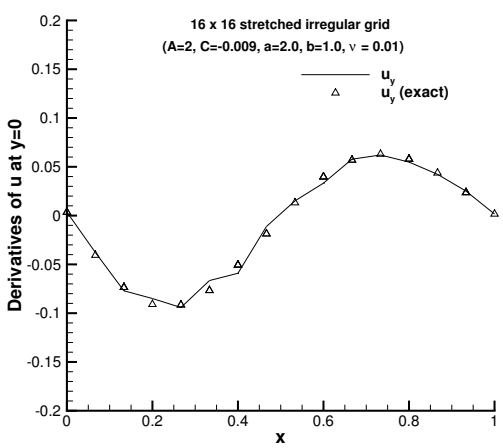
In this section, we verify the accuracy of the RD-CC2 and RD-CC3 schemes for a problem with a curved boundary. We explained and verified in Secs. 5 and 6 that the proposed schemes preserve exact quadratic (RD-CC2) and cubic (RD-CC3) solutions, as designed, for linear (straight-sided) elements. The developed schemes do not require computations and storage of the solution at the edge midpoints; this is the key in obtaining high-order solutions on curved boundaries that are represented with the linear elements. We note that, any modification to the proposed schemes will destroy these special properties (quadratic and cubic exactnesses), and the design order of accuracy of these schemes may not be obtained. Here, we verify the order of accuracy of the solution and solution gradients on a curved geometry for both Dirichlet and Neumann type boundary conditions.



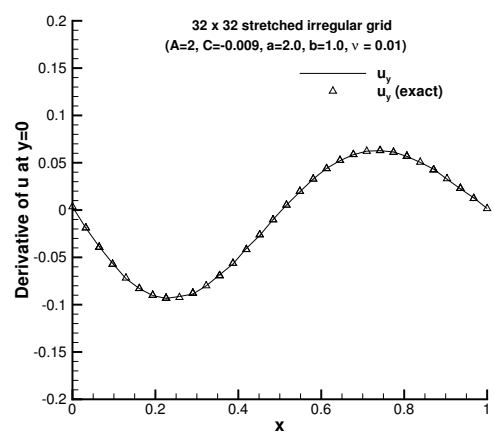
(a) 16×16



(b) 32×32

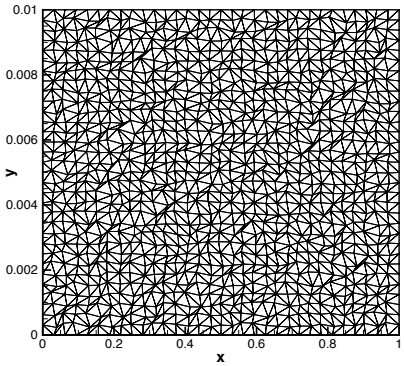


(c) 16×16

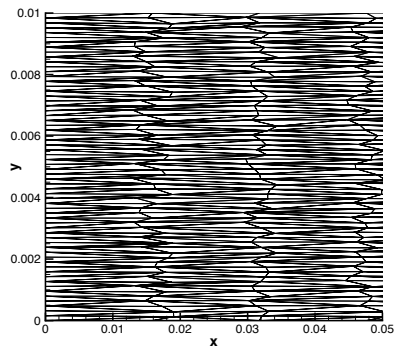


(d) 32×32

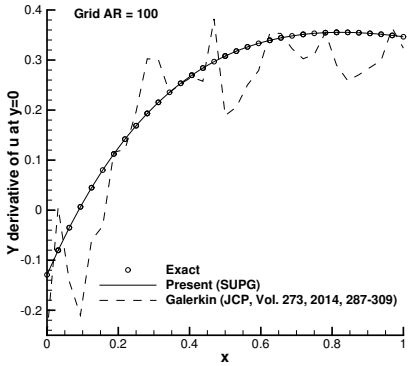
Figure 15: Anisotropic, stretched, and irregular grids, and the corresponding predictions of the normal gradients along the $y = 0$ boundary with the baseline hyperbolic RD scheme ($A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $\nu = 0.01$).



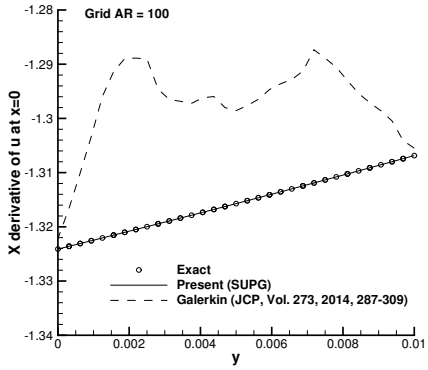
(a) High AR grid (shown with AR=1)



(b) High AR grid (shown with AR=5)



(c) u_y solved on AR=100



(d) u_x solved on AR=100

Figure 16: Comparisons between the gradients (u_x and u_y) predicted with the baseline hyperbolic RD scheme and the published result of the conventional Galerkin method [19] on a highly anisotropic grid with aspect-ratio (AR) 100; ($A = 2$, $C = 1$, $a = 1.23$, $b = 0.12$, $\nu = 0.1235839795$).

For the Neumann type boundary condition, we show that reconstructed high-order normals on the solid boundaries are needed to achieve high-order solution and solution gradients. We point out that the RD-CC3 scheme is not the only such scheme. The third-order edge-based finite-volume scheme has similar properties: it is third-order accurate with the linear elements and requires high-order normals for the Neumann boundary condition as demonstrated in Ref. [26]. We also explain in details the quadratic reconstruction of the boundary normals on the solid surfaces that are represented with the linear elements; the details are given in Appendix A. We then present a high-order integration technique using Simpson's rule, and compare the results with the the low-order trapezoidal rule. A high-order reconstruction technique for estimating the edge midpoint normals and solutions on curved boundaries, which are used in Simpson's rule, is presented in details in Appendix B. The order of accuracy of an integrated quantity on a solid curved boundary is also shown with the solution gradients obtained with the RD-CC2 and the RD-CC3 schemes.

We consider a potential flow problem over a unit circle taken from Ref. [26]. The governing equation is the Laplace equation for the stream function, ψ:

$$\partial_{xx}\psi + \partial_{yy}\psi = 0. \tag{114}$$

As in Ref. [26], we solve the above equation by solving the following hyperbolic system of equations:

$$\partial_\tau \psi = \partial_x p + \partial_y q, \quad \partial_\tau p = \frac{1}{T_r}\left(\partial_x \psi - p\right), \quad \partial_\tau q = \frac{1}{T_r}\left(\partial_y \psi - q\right), \tag{115}$$

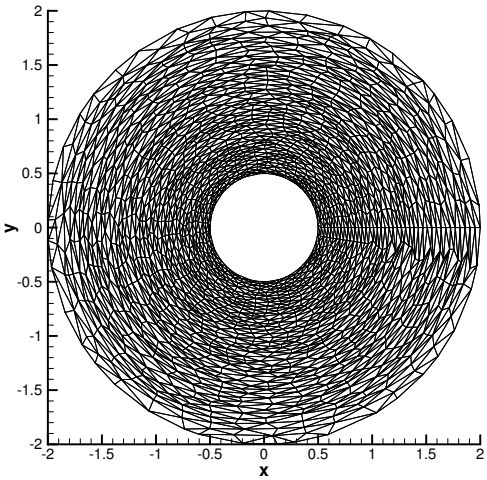
where the x- and y-velocity components are related to the gradient of the main variable ψ; i.e., $\psi_x = -v$ and $\psi_y = u$. The exact solution to this problem is known. See Ref. [32], for example.

We discretize the hyperbolic potential flow system of equations by the RD-CC2 and RD-CC3 schemes, described in Secs. 5 and 6, and verify their predicted orders of accuracy over a set of eight irregular anisotropic triangular grids of 441, 1681, 3721, 6561, 10201, 14641, 25921, and 40401 nodes. The grids are composed of linear elements only, and no curved elements are used. The domain and a sample grid (1681 nodes) is shown in Fig. 17a.

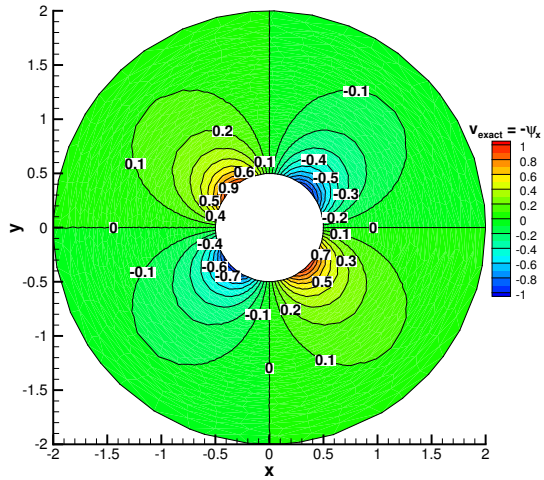
8.2.1. Dirichlet BC

We impose the exact solution on the outer domain, and specify $\psi = 0$ at the solid boundary nodes and predict the gradients at the boundary nodes, and solution and solution gradients in the interior nodes by the numerical schemes. Figure 17 shows the contours of ψ_x obtained with the RD-CC3 scheme on the 1681-node grid. The excellent quality of the predicted derivatives is observed in comparison with the exact solution. The predicted streamline is also shown in the same figure.

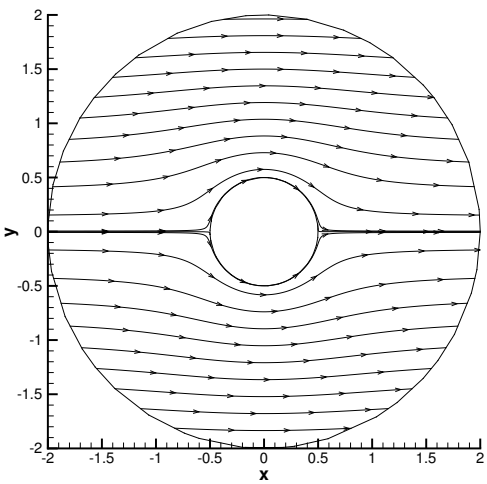
The order of accuracy of theses schemes on predicting solution gradients on curved boundaries is also verified. We obtained the error convergence results by computing the L_2 norms defined as $\mathbf{L}_2(:) = \sqrt{\sum_{j=1}^{N} |\mathbf{U}_j - \mathbf{U}_j^e|^2/N}$, for the solution and the solution gradients. Here, the superscript e denotes the exact value, and N is the total number of nodes. The same convergence rates (slopes 2 and 3) are obtained based on the $\mathbf{L}_1$ error and therefore, not shown. The error convergence results for the boundary nodes and the entire domain are shown in Fig. 18. As it is shown, the design order of accuracy is clearly observed through the boundary nodes. It is also evident that the RD-CC3 scheme is more accurate than the RD-CC2 scheme, even on the coarsest grid level, indicating that the third-order scheme, which is constructed for linear elements, is, in fact, solving the correct problem on the curved boundary that is defined only with linear elements. We note again that the cell residual evaluated with the RD-CC2 and the RD-CC3 schemes vanishes for quadratic and cubic solutions, respectively, on the linear triangular elements, and that is how we can obtain high-order solutions on curved boundaries that are discretized with the linear elements. In fact, curved elements on the boundary will destroy this special property of the proposed schemes, as constructed in the current forms.



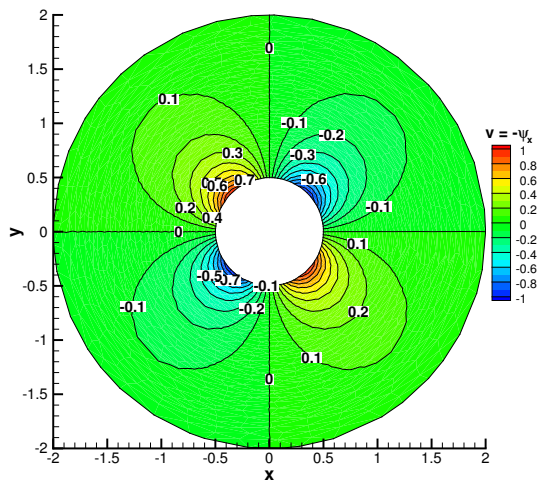
(a) Irregular and perturbed grid (1681 nodes)



(b) Exact $(-\psi_x)$ on the grid of 1681 nodes

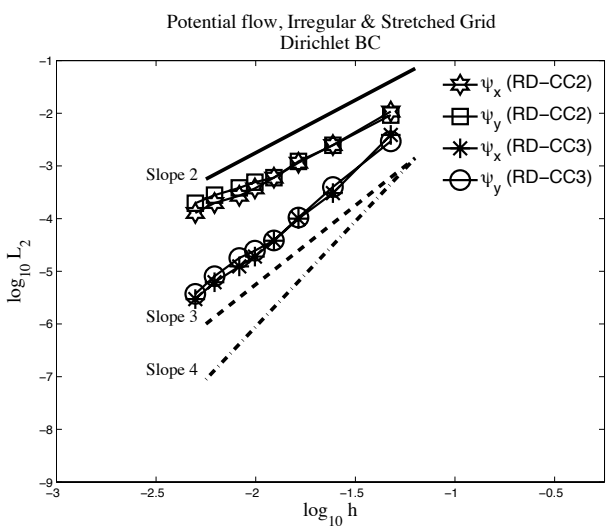


(c) Predicted streamline on the grid of 1681 nodes.

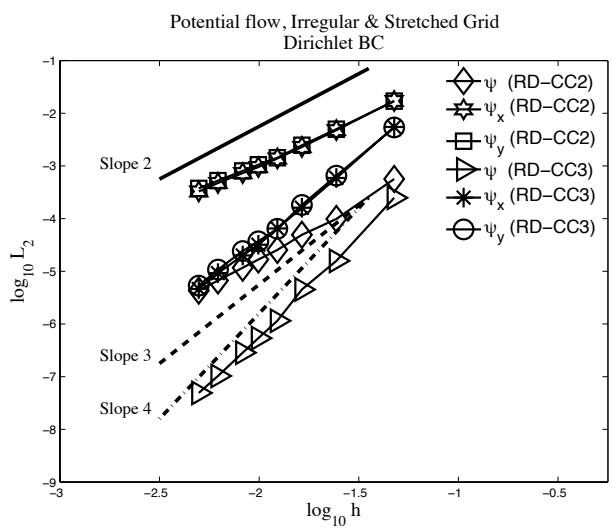


(d) Predicted $(-\psi_x)$ on the grid of 1681 nodes.

Figure 17: Potential flow $(\psi_{xx} + \psi_{yy} = 0)$ over a unit circle on irregular and stretched grids with RD-CC3. The contours show the x-gradient of the solution variable ψ.



(a) Solid boundary (Dirichlet BC)



(b) Entire domain (Dirichlet BC)

Figure 18: Order of accuracy of the RD-CC2 and the RD-CC3 shames for the potential flow $(\psi_{xx} + \psi_{yy} = 0)$ over a unit circle with Dirichlet BC for a series of irregular and stretched grids.

8.2.2. Neumann BC

In this example, we also impose a zero normal velocity on the solid boundary. This is equivalent to the adiabatic wall BC for the Navier-Stokes equations. Therefore, we need to solve the following on the solid boundary (see also Sec. 2.4):

$$\begin{bmatrix} \psi_j - \psi_{j_b} \\ (q_{j_b}, -p_{j_b}) \cdot \tilde{n}_b \\ \mathbf{Res}_{j_b}(3)\tilde{t}_{b_x} - \mathbf{Res}_{j_b}(2)\tilde{t}_{b_y} \end{bmatrix}, \quad \text{if} \quad |\tilde{n}_{b_x}| \leq |\tilde{n}_{b_y}| \tag{116}$$

$$\begin{bmatrix} \psi_j - \psi_{j_b} \\ \mathbf{Res}_{j_b}(3)\tilde{t}_{b_x} - \mathbf{Res}_{j_b}(2)\tilde{t}_{b_y} \\ (q_{j_b}, -p_{j_b}) \cdot \tilde{n}_b \end{bmatrix}, \quad \text{if} \quad |\tilde{n}_{b_x}| > |\tilde{n}_{b_y}| \tag{117}$$

where we have used $u = q$ and $v = -p$ relations. Similarly to the previous test case, we predict the gradients on the boundary nodes, as well as the solution and the solution gradients for the interior nodes by the proposed numerical schemes. We note that the RD-CC3 scheme was designed to preserve exact cubic function (quadratic gradients). Therefore, the curved surface boundary normals must be quadratic to ensure $(q_{j_b}, -p_{j_b}) \cdot \tilde{n}_b$ remains exact for quadratic gradients. We achieve this by locally performing a high-order polynomial fit operation over the curved geometry, represented with straight-sided triangles. We note that the high-order polynomial fit operation over the curved geometry is only local, and only applied to the curved boundary to compute only the surface normals. Thus, curved elements are not needed and therefore, not constructed.

For the third-order edge-based finite-volume scheme that is also designed to be exact for quadratic gradients, it has been demonstrated that high-order normals computed from a quadratic reconstruction of the boundary is sufficient for preserving third-order accuracy at boundary nodes [26]. The details of the quadratic boundary normals reconstruction are provided in Appendix A. Here we only present the final formulations. The reconstructed quadratic boundary normals for node i, $\mathbf{n}_i = (n_{i_x}, n_{i_y})$, are given as:

$$n_{x_i} = \frac{-\Delta s_R}{\Delta s_L\left(\Delta s_L + \Delta s_R\right)} y_1 + \left(\frac{1}{\Delta s_L} - \frac{1}{\Delta s_R}\right) y_2 + \frac{\Delta s_L}{\Delta s_R\left(\Delta s_R + \Delta s_L\right)} y_3, \tag{118}$$

$$n_{y_i} = \frac{\Delta s_R}{\Delta s_L\left(\Delta s_L + \Delta s_R\right)} x_1 - \left(\frac{1}{\Delta s_L} - \frac{1}{\Delta s_R}\right) x_2 - \frac{\Delta s_L}{\Delta s_R\left(\Delta s_R + \Delta s_L\right)} x_3, \tag{119}$$

where Δs_R and Δs_L are the distances between the boundary node i and its immediate right and left boundary node neighbors, respectively. The high-order unit normal vector, $\hat{\mathbf{n}}_i$, is then obtained as:

$$\hat{\mathbf{n}}_i = \frac{\mathbf{n}_i}{|\mathbf{n}_i|}. \tag{120}$$

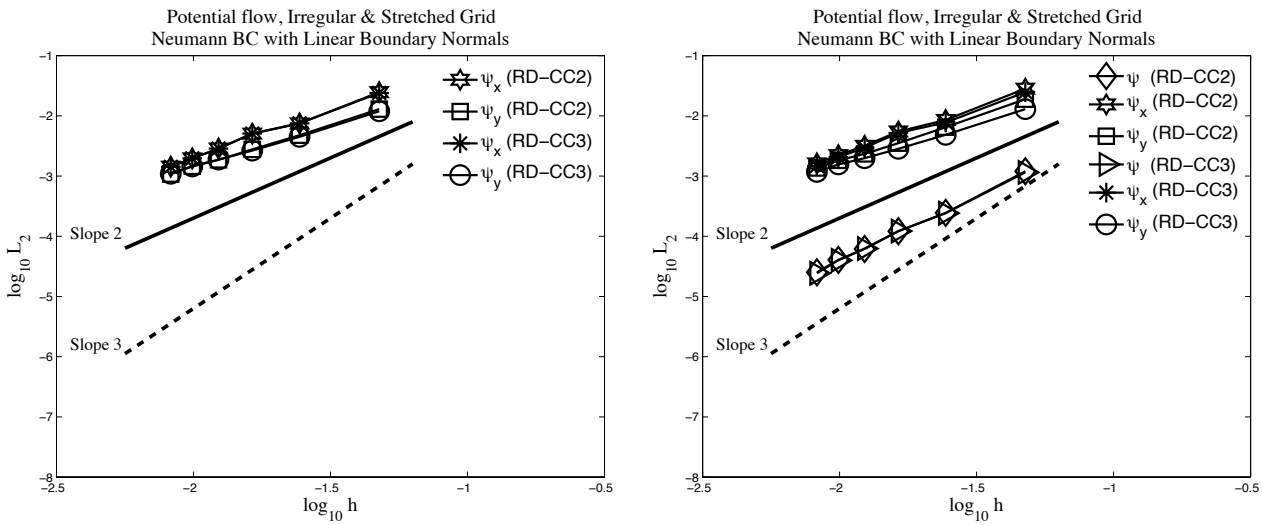
We now solve the discussed potential flow problem over a unit circle with the Neumann BC (zero normal velocity), imposed on the boundary. We implement the Neumann BC with the solid boundary normals that are computed with

1) a linear approximation; i.e., $\mathbf{n}_i = \frac{1}{2}\left(y_{i+1} - y_{i-1},\, x_{i-1} - x_{i+1}\right)$, and

2) a quadratic approximation of the surface boundary normals, as given by Eqs. (118)–(120).

We evaluate the order of accuracy of the proposed RD-CC2 and RD-CC3 schemes and compare the effects of the solid boundary normals on the deign order of accuracy of these schemes. We note that

the quadratic approximation of the unit normals on the cylinder is third-order accurate (cylinder is an exceptional geometry). Therefore, we add $O(h^2)$ to the reconstructed quadratic normals to ensure that the boundary normals are no better than second-order accurate. We obtain the error convergence of the results by computing the L_2 norms as discussed in Sec. 8.2.1. Figure 19 shows the error convergence results with the Neumann BC imposed using the linear approximation of the curved boundary normals for both the RD-CC2 and the RD-CC3 schemes. The same slopes are obtained based on the L_1 errors and therefore, not shown. Order of accuracy of these schemes



(a) Solid boundary (Neumann BC & Linear Normals) (b) Entire domain (Neumann BC & Linear Normals)

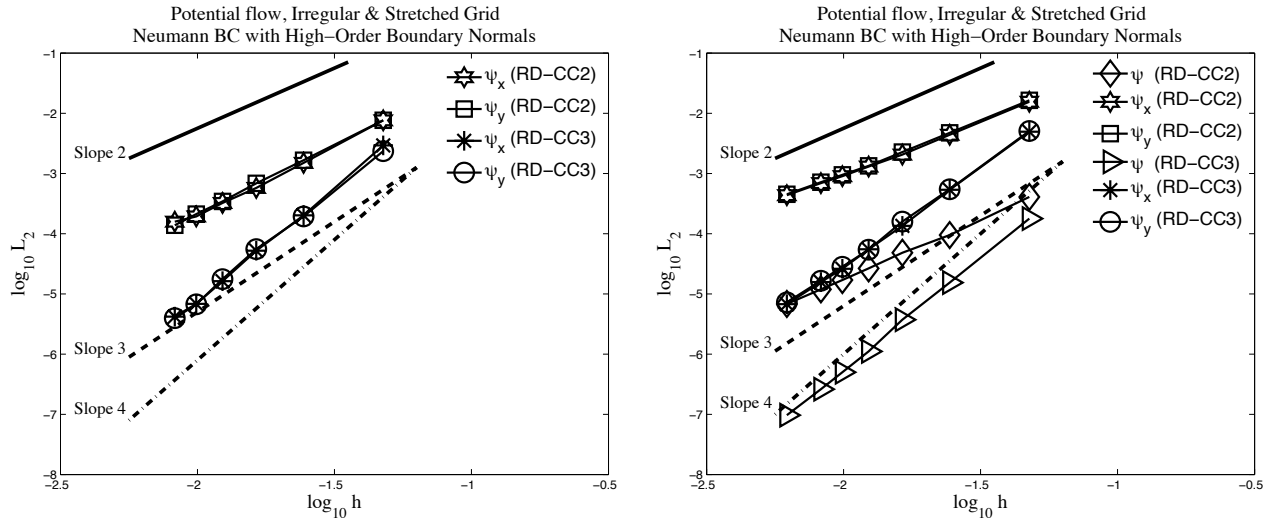
Figure 19: Order of accuracy of the RD-CC2 and the RD-CC3 shames for the potential flow ($\psi_{xx} + \psi_{yy} = 0$) over a unit circle with Neumann BC, imposed on the solid boundary with linear approximation of the boundary normals, on several irregular and stretched grids.

are shown for both the solid curved boundary and the entire computational domain. The results indicate that, imposing the Neumann BC with the linear approximation of the curved boundary normals reduces the accuracy of the third-order RD-CC3 scheme to second-order. In this case, the results predicted by the RD-CC3 scheme becomes almost identical to the results obtained by the RD-CC2 scheme. As expected, the second-order accuracy of the RD-CC2 schemes is unaffected by the linear approximation of the curved surface boundary normals.

As explained earlier, if the quadratic approximation (instead of the linear approximation) of the curved surface boundary normals are employed, the formal design order of accuracy of the third-order RD-CC3 scheme is recovered. These results are shown in Fig. 20. On the other hand, the quadratic approximation of the curved surface boundary normals does not improve the order of accuracy of the RD-CC2 scheme (as expected), but the predicted results are always more accurate than with the linear approximation of the curved surface boundary normals.

8.2.3. Integrated quantity on curved boundaries

In some applications, integrated quantities (such as lift and/or drag) on solid boundaries are of great interest. Accurate and high-order discrete solutions and/or solution gradients are essential in obtaining high-order accurate integrated quantities. In the previous example, we showed that accurate and smooth solution and solution gradients are predicted on curved boundaries with the proposed RD-CC2 and RD-CC3 schemes, without introducing curved elements. Here, we examine the order of accuracy of an integrated value of high-order quantities on a curved boundary, represented with the linear elements. The high-order quantities, in this case, are the high-order solution gradients (i.e., $q = -v$ and $p = u$) obtained with the RD-CC2 and the RD-CC3 schemes for the cylinder case. But, in general, the results of this example are also applicable to high-order quantities that can be produced with any high-order scheme on any curved surface.



(a) Solid boundary (Neumann BC & High-Order Normals) (b) Entire domain (Neumann BC & High-Order Normals)

Figure 20: Order of accuracy of the RD-CC2 and the RD-CC3 schemes for the potential flow ($\psi_{xx} + \psi_{yy} = 0$) over a unit circle with Neumann BC, imposed on the solid boundary with the high-order approximation of the boundary normals, on several irregular and stretched grids.

Consider the following integral on the solid cylinder:

$$\int_{\partial\Omega} \left(u^2 + v^2\right)\, dl \;\; = \;\; \int_{\partial\Omega} \left(q^2 + p^2\right)\, dl, \tag{121}$$

where, $\partial\Omega$ denotes that the integration is performed on the solid boundary, and l is the arc-length. We investigate the impact of the accuracy of the quadrature formula used to evaluate the integral, i.e., whether a high-order quadrature formula is required to achieve third-order accuracy in the integrated quantity. A lower-order quadrature is taken to be the trapezoidal rule as typically employed for second-order schemes. For a high-order quadrature formula, we consider the three-point Simpson rule applied along each edge as described below in details. Note again that this part of the work is not specific to the schemes developed in this paper, but generally applicable to any scheme that can produce third-order accurate gradients on curved boundaries.

The three-point Simpson rule requires estimation of the midpoint location and solutions. For convenience, we apply the Simpson rule in the parameter space of the arc-length. Therefore, we need to estimate the curved surface normal vector and solution values at the midpoint in the parameter coordinate. The details of the midpoint unit normals and solutions reconstruction using the Hermite polynomial functions are given in Appendix B. Here, we only present the final formulations. The reconstructed midpoint normals, $\mathbf{n}_m$, and the unit normals, $\hat{\mathbf{n}}_m$, are

$$n_{x_m} = \frac{3}{2}\frac{y_{i+1} - y_i}{\Delta s_{12}} - \frac{1}{4}\left(\hat{\tau}_{y_i} + \hat{\tau}_{y_{i+1}}\right), \quad n_{y_m} = -\frac{3}{2}\frac{x_{i+1} - x_i}{\Delta s_{12}} + \frac{1}{4}\left(\hat{\tau}_{x_i} + \hat{\tau}_{x_{i+1}}\right), \quad \hat{\mathbf{n}}_m = \frac{\mathbf{n_m}}{|\mathbf{n_m}|}. \tag{122}$$

The midpoint solutions (in this case p_m and q_m) are evaluated similarly (refer to Appendix B for more details), which become:

$$p_m \;\; = \;\; \frac{1}{2}(p_i + p_{i+1}) - \frac{1}{8}\left(p_{i_x}\hat{\tau}_{i_x} + p_{i_y}\hat{\tau}_{i_y} - p_{i+1_x}\hat{\tau}_{i+1_y} - p_{i+1_y}\hat{\tau}_{i+1_y}\right)\Delta s_{12}, \tag{123}$$

$$q_m \;\; = \;\; \frac{1}{2}(q_i + q_{i+1}) - \frac{1}{8}\left(q_{i_x}\hat{\tau}_{i_x} + q_{i_y}\hat{\tau}_{i_y} - q_{i+1_x}\hat{\tau}_{i+1_y} - q_{i+1_y}\hat{\tau}_{i+1_y}\right)\Delta s_{12}, \tag{124}$$

where the gradients of p and q are reconstructed using the quadratic LSQ technique. Note that in our cylinder example, p and q are the gradients of the stream function ψ, and are already available as part of the hyperbolic system formulation.

Finally, we evaluate the high-order integral as

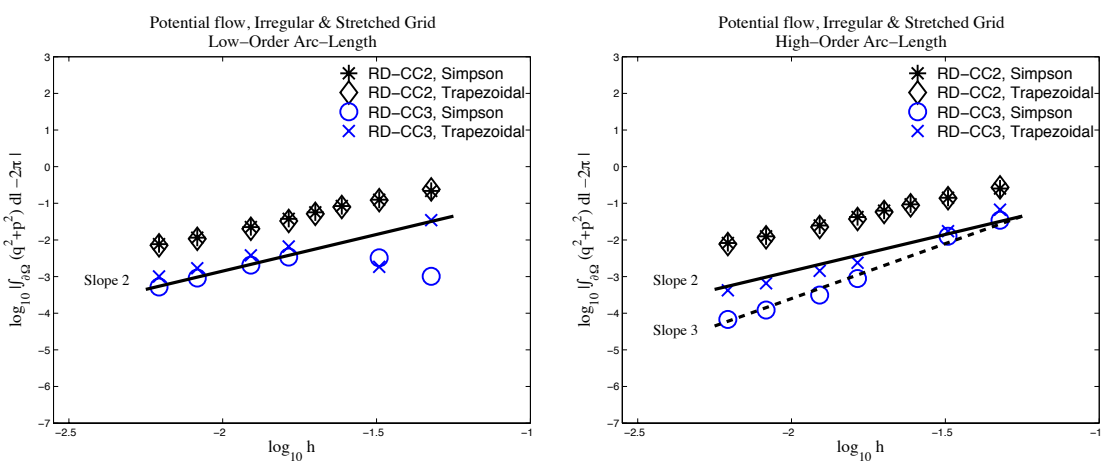
$$\int_{\partial\Omega} \left(u^2+v^2\right)\, dl = \int_{\partial\Omega} \left(q^2+p^2\right)\, dl = \frac{1}{6} \sum_{i\in\{e_b\}} \left[\left(q_i^2+p_i^2\right) + 4\left(q_m^2+p_m^2\right) + \left(q_{i+1}^2+p_{i+1}^2\right)\right] \Delta l_{12}, \quad (125)$$

where $\{e_b\}$ denotes the set of boundary edges over the cylinder. We remark that the accuracy of the arc-length, Δl_{12}, plays a critical role in defining the order of accuracy of an integrated high-order quantity. To examine this, we employ both a low-order and a high-order arc-length estimations. The low-order arc-length is taken as the length of the linear boundary edge Δs_{12}, while the high-order arc-length is estimated with the three-point Simpson rule applied to

$$\int_0^1 \sqrt{\left(\frac{\partial Hx}{\partial \xi}\right)^2 + \left(\frac{\partial Hy}{\partial \xi}\right)^2}\, d\xi, \quad (126)$$

over each linear edge, where Hx and Hy are the reconstructed Hermite polynomial functions used to estimate the edge midpoint solutions (see Appendix B and in particular, Eqs. B.1 and B.2).

We now compute the integral given in Eq. (121) on the solid cylinder using the trapezoidal and the three-point Simpson rules on a series of irregular and perturbed grids with the solutions predicted by both the second-order RD-CC2 and the third-order RD-CC3 schemes. The exact value of this integral for the unit-circle is 2π. For both the low- and the high-order quadratures, we employ the reconstructed quadratic surface normals, Eq. (120). To also examine the effects of the arc-length estimation, we show the results obtained with both the low-order arc-length estimation (i.e., the length of the linear edge) and the high-order reconstructed arc-length (i.e., Eq. 126).



(a) Low-order arc-length (b) High-order arc-length

Figure 21: Computed order of accuracy of $\int_{\partial\Omega}(q^2+p^2)\, dl$ on the solid curved boundary using the second- and third-order solutions obtained with the RD-CC2 and RD-CC3 schemes. The integrations are performed with the quadratic approximation of the curved boundary normals Eq. (122). a) a low-order arc-length estimation is used for the integration, b) Hermite polynomial is fitted along each boundary edge and the high-order arc-length is estimated using Simpson's rule.

The results obtained with the low-order arc-length estimation are shown in Fig. 21(a). The order of accuracy of the integration, Eq. (121), with the low-order arc-length estimation is second-order for either of the second-order or the third-order solution gradients. The trapezoidal and the Simpson rules give almost identical results. The integral obtained with the third-order quantities are generally one-order of magnitude more accurate than the ones obtained with the second-order quantities.

Fig. 21(b) illustrates the oder of accuracy of the integral (Eq. 121) using the high-order estimation of the arc-length (i.e., Eq. 126). The third-order accurate integration on the curved boundary

is achieved for the RD-CC3 scheme using Simpson's rule and the high-order arc-length estimation. Comparing this result with the one presented in Fig. 21(s), we clearly see that the accuracy of the curved boundary arc-length is the dominant parameter for high-order integrations. The order of accuracy of the integration is reduced to second-order when, instead, the trapezoidal rule is employed. For the second-order quantities (in this case the solution gradients p and q produced by the RD-CC2 scheme), the high-order arc-length estimation neither improve the order of accuracy of the integration nor provide a more accurate estimate for the integral. Note that the solution gradients, which are predicted with the RD-CC2 and the RD-CC3 schemes, and used in the integration, are smooth. It is of interest, also, to study the effects of the high-order solutions and the solution gradients predicted with conventional schemes (with noisy and inaccurate solution gradients) using similar irregular grids on the integrated quantities. This is beyond the scope of the present study and therefore, is left for future studies. Finally, these results demonstrate clearly that high-order grids with curved elements are not required for achieving third-order accuracy in the solution (and the gradients) as well as in integral quantities. A high-order reconstruction of a curved geometry is sufficient to achieve third-order accuracy.

8.3. Nonlinear advection-diffusion problem

Consider the following nonlinear advection-diffusion equation:

$$\partial_t u + \partial_x f + \partial_y g = \partial_x(\nu \partial_x u) + \partial_y(\nu \partial_y u) + \tilde{s}(x, y), \tag{127}$$

where $f = g = u^2/2$, $\nu = u$. The source term $\tilde{s}(x, y)$ is defined by

$$\tilde{s}(x, y) = u^e(u^e_x + u^e_y) - (u^e_x)^2 - (u^e_y)^2 - u^e(u^e_{xx} + u^e_{yy}), \tag{128}$$

$$u^e = C\cos(A\pi\eta)\exp\left(\frac{1 - \sqrt{1 + 4A^2\pi^2\nu_0^2}}{2\nu_0}\xi\right) + C_0, \tag{129}$$

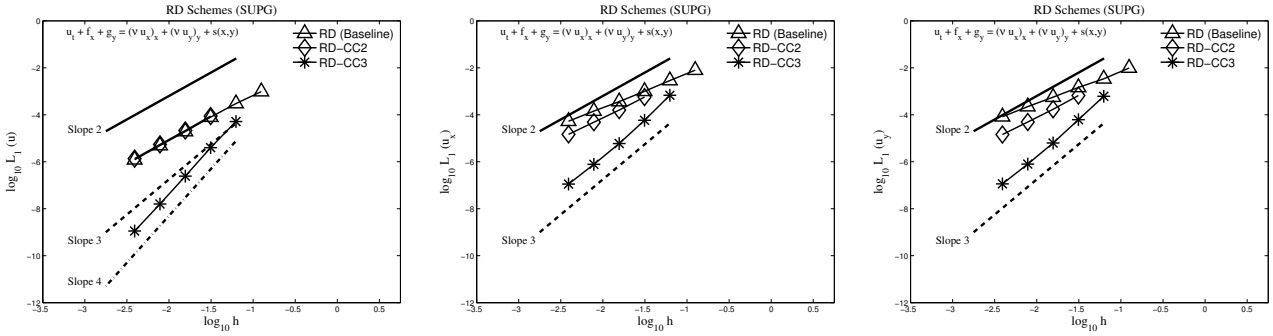
where $C = -0.009$, $A = 2.0$, and $C_0 = 1$, so that u^e is the exact solution to the nonlinear advection-diffusion equation (127). Note that C_0 must be greater than C in order for the diffusion coefficient to be positive.

We solve the nonlinear advection-diffusion equation in a square domain of [0,1]×[0,1], and verify the order of accuracy of the solution and solution gradients predicted by the baseline RD, the RD-CC2, and the RD-CC3 schemes. The order of accuracy comparison is shown in Fig. 22 along with a more detailed information in Tables 9–11, illustrating that the RD-CC2 and RD-CC3 schemes achieve an improved order of accuracy for both the solution and the solution gradients on irregular grids. On the other hand, the baseline RD scheme suffers from the reduced order of accuracy in the gradients, as well as noisy predicted solution gradients as discussed in Sec. 8.1.

Table 9: L_1 error convergence of u for the hyperbolic RD schemes with the SUPG distribution on irregular grids, for the nonlinear advection-diffusion problem ($f = g = u^2/2$, $\nu = u$, $\nu_0 = 0.01$).

Grids	RD	Order	RD-CC2	Order	RD-CC3	Order
32×32	7.93E-05	–	8.79E-05	–	3.98E-06	–
64×64	1.93E-05	2.04	2.15E-05	2.03	2.43E-07	4.04
128×128	4.90E-06	1.98	5.46E-06	1.98	1.59E-08	3.93
256×256	1.22E-06	2.01	1.36E-06	2.01	1.12E-09	3.83

Order of accuracy of the proposed schemes, computed for this nonlinear problem across a wide range of $Re = a/\nu_0$, ($\nu_0 = 10^{+2}\ldots 10^{-8}$), are also obtained and shown in Fig. 23. This figure, which compares the results for the baseline RD, and the proposed RD-CC2 and RD-CC3 schemes, indicate that the design order of accuracy is obtained for all the Re cases. These cases are converged



(a) u (b) u_x (c) u_y

Figure 22: Order of accuracy comparisons between the baseline and the proposed hyperbolic RD schemes on irregular grids for the nonlinear advection-diffusion problem ($f = g = u^2/2$, $\nu = u$, $\nu_0 = 0.01$).

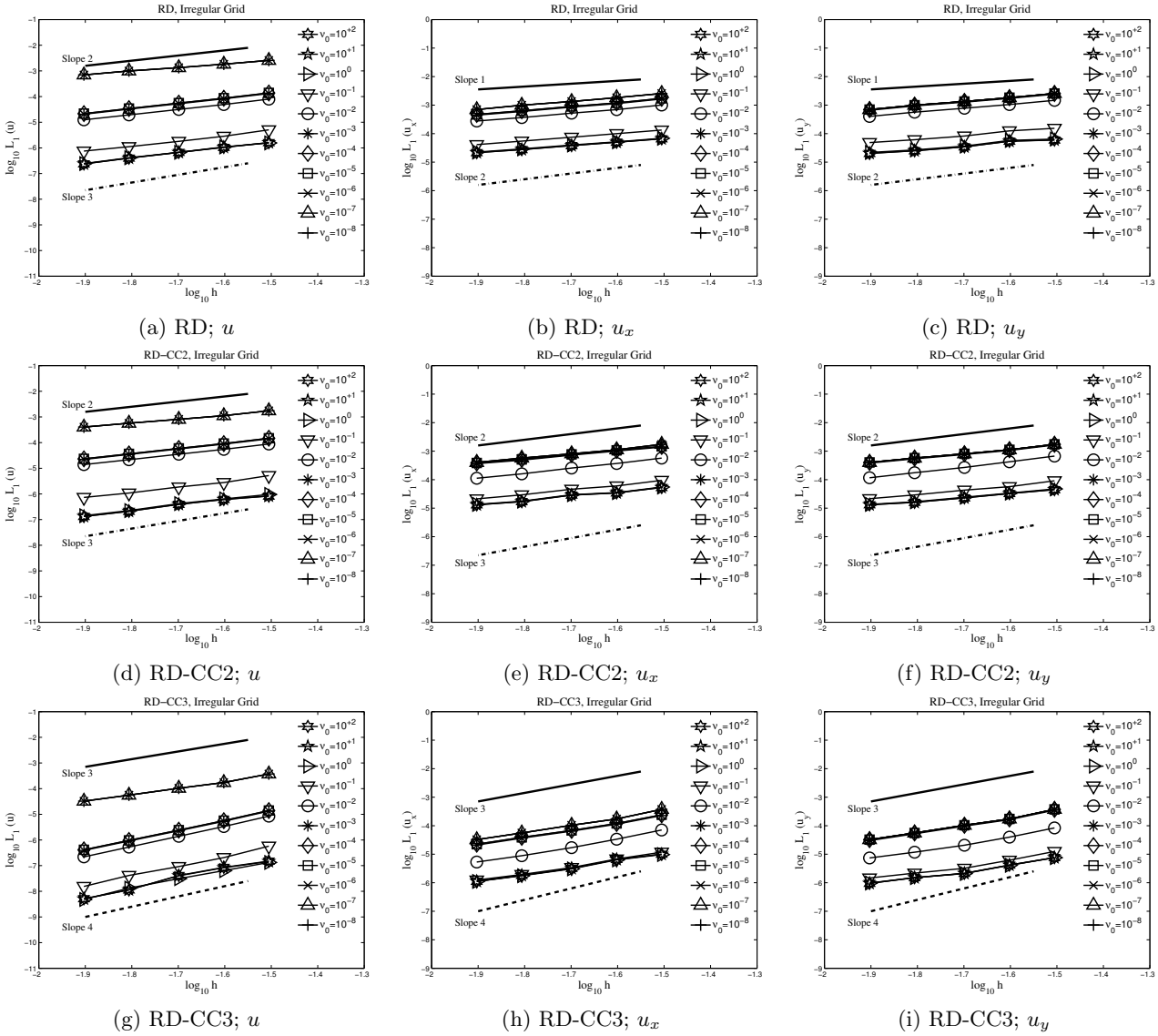
Table 10: L_1 error convergence of u_x for the hyperbolic RD schemes with the SUPG distribution on irregular grids, for the nonlinear advection-diffusion problem ($f = g = u^2/2$, $\nu = u$, $\nu_0 = 0.01$).

Grids	RD	Order	RD-CC2	Order	RD-CC3	Order
32×32	1.00E-03	–	5.66E-04	–	5.84E-05	–
64×64	3.63E-04	1.46	1.59E-04	1.83	5.94E-06	3.30
128×128	1.44E-04	1.33	4.84E-05	1.72	7.75E-07	2.94
256×256	5.31E-05	1.44	1.47E-05	1.72	1.12E-07	2.79

Table 11: L_1 error convergence of u_y for the hyperbolic RD schemes with the SUPG distribution on irregular grids, for the nonlinear advection-diffusion problem ($f = g = u^2/2$, $\nu = u$, $\nu_0 = 0.01$).

Grids	RD	Order	RD-CC2	Order	RD-CC3	Order
32×32	1.46E-03	–	6.59E-04	–	6.17E-05	–
64×64	5.62E-04	1.38	1.71E-04	1.95	6.28E-06	3.30
128×128	2.23E-04	1.33	4.91E-05	1.80	7.94E-07	2.98
256×256	8.23E-05	1.44	1.44E-05	1.77	1.14E-07	2.80

to $\leq 10^{-10}$ with less than 15 Newton iterations and about 100–300 Gauss-Seidel relaxations per Newton iteration. Figure 24 illustrates that the average number of Gauss-Seidel relaxations per Newton iteration increases linearly with grid refinements for all the Re cases.



(a) RD; u (b) RD; u_x (c) RD; u_y

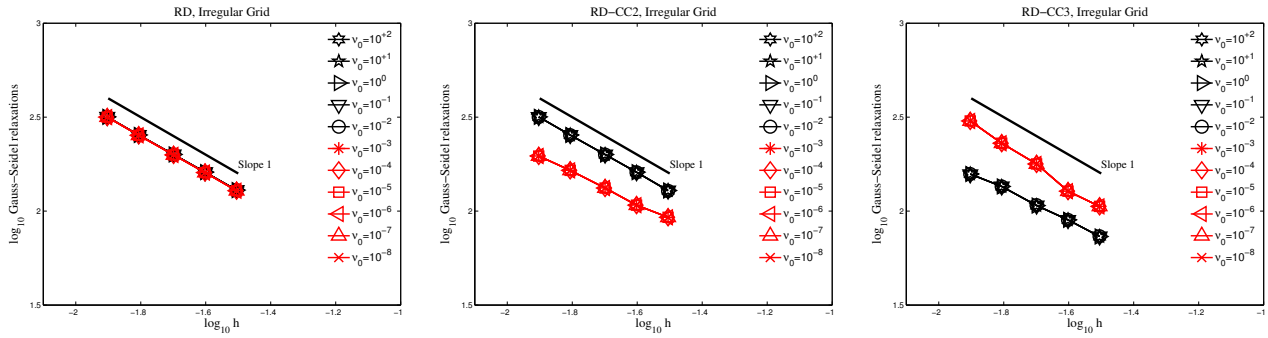
(d) RD-CC2; u (e) RD-CC2; u_x (f) RD-CC2; u_y

(g) RD-CC3; u (h) RD-CC3; u_x (i) RD-CC3; u_y

Figure 23: Uniform accuracy for the nonlinear problem over a range of Re ($\nu_0 = 10^{+2} \ldots 10^{-8}$) with the RD and the proposed RD-CC2 and RD-CC3 schemes on irregular grids ($f = g = u^2/2$, $\nu = u$, $A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $Re = \sqrt{5}/\nu_0$).

9. Conclusions

We have presented detailed formulation and implementation procedures for hyperbolic RD schemes for general advection-diffusion problems on arbitrary triangular grids. Improving upon the previous work [16, 17], we developed new second- and third-order hyperbolic RD schemes that preserve, respectively, quadratic and cubic solutions on arbitrary triangular grids. We also developed a high-order source term discretization, which was found to be essential in constructing the third-order scheme with linear elements. We showed that the second- and third-order schemes can be constructed by a separate treatment of the advective and diffusive terms, enabling development of hyperbolic residual-distribution schemes for the compressible Navier-Stokes equations. The superiority of the proposed schemes was further demonstrated with accurate and smooth predictions of



(a) RD, GS relaxations (b) RD-CC2, GS relaxations (c) RD-CC3, GS relaxations

Figure 24: Average Gauss-Seidel relaxations per Newton iteration (typically less than 15 Newton iterations) performed by the proposed RD-CC2 and RD-CC3 schemes for the nonlinear problem over a range of Re ($\nu_0 = 10^{+2} \ldots 10^{-8}$) on a series of irregular grids ($f = g = u^2/2$, $\nu = u$, $A = 2$, $C = -0.009$, $a = 2$, $b = 1$, $Re = \sqrt{5}/\nu_0$).

the gradients within domains, on both flat and curved boundaries. The new second-order scheme, which is still defined within a compact stencil, allows an efficient construction of Newton's method with the exact residual Jacobian computed by the automatic differentiation. We demonstrated that the resulting implicit solver is capable of reducing the residuals of all equations by at least ten orders of magnitude with typically less than 10–15 Newton iterations and an average of 200–800 linear relaxations per Newton iteration. Because of the hyperbolic reformulation of the advection-diffusion equation, the number of linear relaxations per Newton iteration was also shown to increase linearly, not quadratically as for typical diffusion problems, with grid refinement. We also showed that the proposed schemes do not require curved elements for geometries containing curved boundaries, because the cell residual of the RD-CC2 and RD-CC3 schemes, as designed, vanishes for exact quadratic and cubic solutions and fluxes on the linear elements. For completeness, we also presented a high-order integration technique for curved geometries that are represented with the linear elements.

Acknowledgments

The authors would like to thank the Center Chief Technology Office of NASA Langley Research Center for their support through the Center Innovation Fund (CIF) project. The second author was also partially supported by the U.S. Army Research Office under the Contract/Grant number W911NF-12-1-0154. The authors would also like to thank the anonymous reviewers for their constructive comments.

Appendix A. High-order reconstruction of curved boundary normals

Here, we explain in details a quadratic reconstruction of the straight boundary edges for curved geometries as employed in this work and in Ref. [26]. This reconstruction strategy is used to compute high-order boundary normals for curved geometries that are discretized with the linear edges. We need the quadratic curved boundary normals for a Neumann type BC implementation discussed in Sec. 2.4. Note that the presented quadratic approximation of the curved boundary normals is constructed from the existing boundary nodes with straight-sided edges and therefore, does not provide an exact normal vector for the curved boundaries.

Consider the boundary nodes on the curved geometry represented with the linear (straight-sided) elements (Fig. A.25). We reconstruct a quadratic function in the parameter space of the arc-length for the three consecutive boundary nodes $i-1$, i, and $i+1$, to obtain the high-order normal at the boundary node i. We start the procedure by assuming that x and y coordinates of

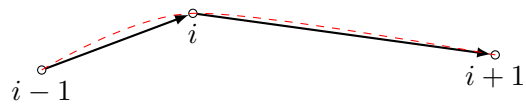


Figure A.25: Schematic of boundary nodes connected with straight-sided edges (black), and with a reconstructed quadratic edges (red).

the boundary, which must pass through the three consecutive boundary nodes $i-1$, i, and $i+1$, can be expressed as:

$$Fx(r) = x_{i-1}\,Q_{i-1}(r) + x_i\,Q_i(r) + x_{i+1}\,Q_{i+1}(r), \tag{A.1}$$

$$Fy(r) = y_{i-1}\,Q_{i-1}(r) + y_i\,Q_i(r) + y_{i+1}\,Q_{i+1}(r), \tag{A.2}$$

where $Q_k(r)$ are the quadratic basis functions in the parameter space r:

$$Q_k(r) = a_k + b_k r + c_k r^2, \quad k = i-1,\, i,\, i+1, \tag{A.3}$$

and a_k, b_k, and c_k are constants that need to be defined. Note that, we take the arc-length of the boundary edges as the parameter r measured from the boundary node $i-1$ to the boundary node $i+1$, along the straight-sided edges; i.e.,

$$r = \begin{cases} 0 & \text{at node } i-1 \\ \Delta s_L & \text{at node } i \\ \Delta s_L + \Delta s_R & \text{at node } i+1 \end{cases}, \tag{A.4}$$

where Δs_R and Δs_L are the distances between the boundary node i and its immediate right and left boundary node neighbors, respectively.

The constants of the quadratic basis functions are found by noting that the reconstructed quadratic function passes through the three consecutive boundary nodes; i.e.,

$$Fx(0) = x_{i-1}, \qquad Fx(\Delta s_L) = x_i, \qquad Fx(\Delta s_L + \Delta s_R) = x_{i+1}, \tag{A.5}$$

$$Fy(0) = y_{i-1}, \qquad Fy(\Delta s_L) = y_i, \qquad Fy(\Delta s_L + \Delta s_R) = y_{i+1}, \tag{A.6}$$

which translates to the following constraints for the three quadratic basis functions:

$$Q_{i-1}(0) = 1, \qquad Q_{i-1}(\Delta s_L) = 0, \qquad Q_{i-1}(\Delta s_L + \Delta s_R) = 0, \tag{A.7}$$

$$Q_i(0) = 0, \qquad Q_i(\Delta s_L) = 1, \qquad Q_i(\Delta s_L + \Delta s_R) = 0, \tag{A.8}$$

$$Q_{i+1}(0) = 0, \qquad Q_{i+1}(\Delta s_L) = 0, \qquad Q_{i+1}(\Delta s_L + \Delta s_R) = 1. \tag{A.9}$$

Solving these constraints for the coefficients, we obtain

$$a_{i-1} = 1, \qquad b_{i-1} = -\frac{2\Delta s_L + \Delta s_R}{\Delta s_L(\Delta s_L + \Delta s_R)}, \qquad c_{i-1} = \frac{1}{\Delta s_L(\Delta s_L + \Delta s_R)}, \tag{A.10}$$

$$a_i = 0, \qquad b_i = \frac{\Delta s_L + \Delta s_R}{\Delta s_L \Delta s_R}, \qquad c_i = -\frac{1}{\Delta s_L \Delta s_R}, \tag{A.11}$$

$$a_{i+1} = 0, \qquad b_{i+1} = -\frac{\Delta s_L}{\Delta s_R(\Delta s_L + \Delta s_R)}, \qquad c_{i+1} = \frac{1}{\Delta s_R(\Delta s_L + \Delta s_R)}. \tag{A.12}$$

With the full description of the quadratic basis functions defined in the parameter space r, we can compute the high-order tangent vector at the boundary node i, τ_i, by differentiating the reconstructed quadratic functions, Eqs. (A.1) and (A.2), at $r = \Delta s_L$:

$$\tau_i = \left(\frac{d\,Fx(r)}{dr}, \frac{d\,Fy(r)}{dr} \right) |_{r=\Delta s_L}. \tag{A.13}$$

where, after some algebra, we arrive at:

$$\tau_{x_i} = \frac{-\Delta s_R}{\Delta s_L\,(\Delta s_L + \Delta s_R)} x_1 + \left(\frac{1}{\Delta s_L} - \frac{1}{\Delta s_R} \right) x_2 + \frac{\Delta s_L}{\Delta s_R\,(\Delta s_R + \Delta s_L)} x_3, \tag{A.14}$$

$$\tau_{y_i} = \frac{-\Delta s_R}{\Delta s_L\,(\Delta s_L + \Delta s_R)} y_1 + \left(\frac{1}{\Delta s_L} - \frac{1}{\Delta s_R} \right) y_2 + \frac{\Delta s_L}{\Delta s_R\,(\Delta s_R + \Delta s_L)} y_3. \tag{A.15}$$

The high-order normal vector, $\mathbf{n}_i$, and also the unit vector, $\hat{\mathbf{n}}_i$, are then obtained by rotating the tangent vector; i.e.,

$$\mathbf{n}_i = (\tau_{y_i}, -\tau_{x_i}), \quad \hat{\mathbf{n}}_i = \frac{\mathbf{n}_i}{|\mathbf{n}_i|}. \tag{A.16}$$

Appendix B. High-order reconstruction of the edge midpoint normals and solutions

Here we present a Hermite polynomial reconstruction technique to estimate the boundary edge midpoint normals and solutions for curved boundaries that are represented with the linear (straight-sided) elements. In this present study, the boundary edge normals and solutions are used for high-order integration on solid curved boundaries with the three-point Simpson rule. Note that the presented technique is not restricted to the proposed high-order schemes, and applicable to any scheme that can produce high-order solution and/or solution gradients on the curved boundaries.

The midpoint normals are computed by defining the Hermite interpolating functions in the parameter space $\xi = (r - r_i)/(r_{i+1} - r_i) \in (0, 1)$:

$$Hx(\xi) = (2\xi^3 - 3\xi^2 + 1)x_i + (-2\xi^3 + 3\xi^2)x_{i+1} + (\xi^3 - 2\xi^2 + \xi)\hat{\tau}_{x_i}\Delta s_{12} + (\xi^3 - \xi^2)\hat{\tau}_{i+1_x}\Delta s_{12}, \tag{B.1}$$

$$Hy(\xi) = (2\xi^3 - 3\xi^2 + 1)y_i + (-2\xi^3 + 3\xi^2)y_{i+1} + (\xi^3 - 2\xi^2 + \xi)\hat{\tau}_{y_i}\Delta s_{12} + (\xi^3 - \xi^2)\hat{\tau}_{i+1_y}\Delta s_{12}, \tag{B.2}$$

where r is the arc-length parameter, $r_i = 0$, $r_{i+1} = \Delta s_{12}$, Δs_{12} is the length of the straight boundary edge, and $\hat{\tau}_i$ and $\hat{\tau}_{i+1}$ are the unit tangent vectors, $\tau/|\tau|$, at the boundary nodes i and $i+1$, respectively. Note, ξ is zero at the boundary node i and one at the boundary node $i+1$. We then differentiate the Hermite interpolating functions w.r.t. the arc-length parameter r at $\xi = 1/2$ (i.e., the midpoint), and obtain the midpoint tangent vectors, τ_m:

$$\tau_{x_m} = \frac{3}{2}\frac{x_{i+1} - x_i}{\Delta s_{12}} - \frac{1}{4}\left(\hat{\tau}_{x_i} + \hat{\tau}_{x_{i+1}}\right), \quad \tau_{y_m} = \frac{3}{2}\frac{y_{i+1} - y_i}{\Delta s_{12}} - \frac{1}{4}\left(\hat{\tau}_{y_i} + \hat{\tau}_{y_{i+1}}\right) \tag{B.3}$$

The midpoint unit normals, $\mathbf{n}_m$ can be found by rotating the unit tangent vectors; i.e., $\hat{n}_{x_m} = \hat{\tau}_{y_m}$, $\hat{n}_{y_m} = -\hat{\tau}_{x_m}$.

The solution or the solution gradients at the midpoints is evaluated similarly, by replacing x or y in the Hermite interpolation functions with $\tilde{u}$, and $\hat{\tau}_x$ or $\hat{\tau}_y$ with $d\tilde{u}/dr$. We then set $\xi = 1/2$ (i.e., the midpoint), and arrive at

$$\tilde{u}_m = \frac{1}{2}(\tilde{u}_i + \tilde{u}_{i+1}) - \frac{1}{8}\left(\frac{d\tilde{u}}{dr}|_{i+1} - \frac{d\tilde{u}}{dr}|_i \right) \Delta s_{12}, \tag{B.4}$$

which is the same formula given by Eq. (61), but now expressed in the arc-length parameter r. Using the chain rule, we have

$$\frac{d\tilde{u}}{dr} = \frac{d\tilde{u}}{dx}\frac{dx}{dr} + \frac{d\tilde{u}}{dy}\frac{dy}{dr} = (\tilde{u}_x, \tilde{u}_y) \cdot (\tau_x, \tau_y). \tag{B.5}$$

Substituting Eq. (B.5) into Eq. (B.4), we arrive at

$$\tilde{u}_m = \frac{1}{2}(\tilde{u}_i + \tilde{u}_{i+1}) - \frac{1}{8}\left(\tilde{u}_{x_i}\hat{\tau}_{x_i} + \tilde{u}_{y_i}\hat{\tau}_{y_i} - \tilde{u}_{x_{i+1}}\hat{\tau}_{y_{i+1}} - \tilde{u}_{y_{i+1}}\hat{\tau}_{y_{i+1}}\right)\Delta s_{12}. \tag{B.6}$$

We note that if $\tilde{u}$ denotes the solution (not the solution gradients), $\tilde{u}_x$ and $\tilde{u}_y$ are then the solution gradients, which are already available as part of the hyperbolic system formulation and therefore no LSQ reconstruction is needed. On the other hand, quadratic LSQ reconstructions of $\tilde{u}_x$ and $\tilde{u}_y$ variables are needed if $\tilde{u}$ represents the solution gradients.

[1] P.A. Gnoffo. Multi-dimensional, inviscid flux reconstruction for simulation of hypersonic heating on tetrahedral grids. In *Proc. of 47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition*, AIAA Paper 2009–599, Orlando, Florida, 2009.

[2] K. Kitamura, E. Shima, Y. Nakamura, and P.L. Roe. Evaluation of euler fluxes for hypersonic heating computations. *AIAA J.*, 48(4):763–776, 2010.

[3] Z.J. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H.T. Huynh, N. Kroll, G. May, P.-O Persson, B. Van Leer, and M. Visbal. High–order CFD methods: Current status and perspective. *Int. J. Numer. Meth. Fluids*, 72:811–845, 2012.

[4] M.A. Park, V.N. Vatsa, M.R. Khorram, and D.P. Lockard. Aeroacoustic simulation of nose landing gear on adaptive unstructured grids with FUN3D. In *Proc. of 19th AIAA/CEAS Aeroacoustics Conference*, AIAA Paper 2013–2071, Berlin, Germany, 2011.

[5] G. Rossiello, P. De Palma, G. Pascazio, and M. Napolitano. Third-order-accurate fluctuation-splitting schemes for unsteady hyperbolic problems. *J. Comput. Phys.*, 222:332–352, 2007.

[6] M. Ricchiuto, N. Villedieu, R. Abgrall, and H. Deconinck. On uniformly high-order accurate residual distribution schemes for advection-diffusion. *J. Comput. Appl. Math.*, 215:547–556, 2008.

[7] D. Caraeni and L. Fuchs. Compact third-order multidimensional upwind scheme for Navier-Stokes simulations. *Appl. Math.*, 15:373–401, 2002.

[8] H. Nishikawa, M. Rad, and P. Roe. A third-order fluctuation-splitting scheme that preserves potential flow. In *Proc. of 15th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 01–2595, Anaheim, CA, 2001.

[9] H. Nishikawa and P.L. Roe. On high-order fluctuation-splitting schemes for Navier-Stokes equations. In C. Groth and D.W. Zingg, editors, *Computational Fluid Dynamics 2004*, pages 799–804. Springer-Verlag, 2004.

[10] R. Abgrall and D. De Santis. Linear and non-linear high order accurate residual distribution schemes for the discretization of the steady compressible navierstokes equations. *J. Comput. Phys.*, 283:329–359, 2015.

[11] R. Abgrall. A review of residual distribution schemes for hyperbolic and parabolic problems: the July 2010 state of the art. *Commun. Comput. Phys.*, 11(4):1043–1080, 2012.

[12] R. Abgrall, D. De Santis, and M. Ricchiuto. High order residual distribution scheme for the RANS equations. In *Proc. of 7th International Conference on Computational Fluid Dynamics (ICCFD7)*, ICCFD7-2802, Big Island, Hawaii, July 9–13 2012.

[13] R. Abgrall, P.M. Congedo, D. De Santis, and N. Razzaly. A non-linear residual distribution scheme for real-gas computations. *Computer and Fluids*, 102:148–169, 2014.

[14] Dante De Santis. *Development of a high-order residual distrbition method for Navier-Stokes and RANS equation*. PhD thesis, Institut de Mathematiques, Universite de Bordeaux, 2013.

[15] R. Abgrall, D. De Santis, and M. Ricchiuto. High-order preserving residual distribution schemes for advection-diffusion scalar problems on arbitrary grids. *SIAM J. Sci. Comput.*, 36(3):A955–A983, 2014.

[16] H. Nishikawa. A first-order system approach for diffusion equation. I: Second order residual distribution schemes. *J. Comput. Phys.*, 227:315–352, 2007.

[17] H. Nishikawa. A first-order system approach for diffusion equation. II: Unification of advection and diffusion. *J. Comput. Phys.*, 229:3989–4016, 2010.

[18] H. Nishikawa. New-generation hyperbolic Navier-Stokes schemes: $O(1/h)$ speed-up and accurate viscous/heat fluxes. In *Proc. of 20th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2011–3043, Honolulu, Hawaii, 2011.

[19] H. Nishikawa. First-, second-, and third-order finite-volume schemes for diffusion. *J. Comput. Phys.*, 256:791–805, 2014.

[20] H. Nishikawa. First, second, and third order finite-volume schemes for advection-diffusion. *J. Comput. Phys.*, 273:287–309, 2014.

[21] H. Nishikawa. First, second, and third order finite-volume schemes for Navier-Stokes equations. In *Proc. of 7th AIAA Theoretical Fluid Mechanics Conference, AIAA Aviation and Aeronautics Forum and Exposition 2014*, AIAA Paper 2014–2091, Atlanta, GA, 2014.

[22] H. Nishikawa and P.L. Roe. Active flux for diffusion. In *Proc. of 7th AIAA Theoretical Fluid Mechanics Conference, AIAA Aviation and Aeronautics Forum and Exposition 2014*, AIAA Paper 2014-2092, Atlanta, GA, 2014.

[23] A. Mazaheri and H. Nishikawa. First-order hyperbolic system method for time-dependent advection-diffusion problems. Technical Report TM–2014–218175, NASA, March 2014.

[24] A. Mazaheri and H. Nishikawa. Very efficient high-order hyperbolic schemes for time-dependent advection-diffusion problems: Third-, fourth-, and sixth-order. *Computers and Fluids*, 102:131–147, 2014.

[25] L. Krivodonova and M. Berger. High-order accurate implementation of solid wall boundary conditions in curved geometries. *J. Comput. Phys.*, 211:492–512, 2006.

[26] H. Nishikawa. Accuracy-preserving boundary flux quadrature for finite-volume discretization on unstructured grids. *J. Comput. Phys.*, 281:518–555, 2015.

[27] H. Nishikawa. Multigrid third-order least-squares solution of Cauchy-Riemann equations on unstructured triangular grids. *Int. J. Numer. Meth. Fluids*, 53:443–454, 2007.

[28] A. Mazaheri and H. Nishikawa. High-order hyperbolic residual-distribution schemes on arbitrary triangular grids. In *Proc. of , 22nd AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2015–2445, Dallas, Texas, 2015.

[29] A. Brooks and T.J.R. Hughes. Streamline upwind petrov-galerkin formulation for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.*, 32:199–259, 1982.

[30] A. De Palma, G. Pascazio, D.T. Rubino, and M. Napolitano. Redidual-distribution schemes for advection and advection-diffusion problems on quadrilateral cells. *J. Comput. Phys.*, 218:159–199, 2006.

[31] D. Caraeni and L. Fuchs. A new compact high order multidimensional upwind discretization. In *Proc. of 4th World CSCC conference*, Vouliagmeni, Greece, July 10–15 2000.

[32] Katate Masatsuka. *I do like CFD, VOL.1, Second Edition*. Self published, 2013.

[33] R. Abgrall. Toward the ultimate conservative scheme: following the quest. *J. Comput. Phys.*, 167:277–315, 2001.

[34] R. Abgrall and P.L. Roe. High-order fluctuation schemes on triangular meshes. *J. Sci. Comput.*, 19:3–36, 2003.

[35] Hiroaki Nishikawa. Higher-order discretization of diffusion terms in residual-distribution methods. In *34th VKI CFD Lecture Series Very-High Order Discretization Methods*. VKI Lecture Series, 2005.

[36] Li Wang, W. Kyle Anderson, J. Taylor Erwin, and Sagar Kapadia. Discontinuous galerkin and petrov galerkin methods for compressible viscous flows. *Computers and Fluids*, 100:13–29, 2014.