

Beyond Interface Gradient: A General Principle for Constructing Diffusion Schemes

Hiroaki Nishikawa*

National Institute of Aerospace, Hampton, VA 23666, USA

In this paper, we introduce a general principle for constructing time-accurate diffusion schemes, which is applicable to various discretization methods, including finite-volume, residual-distribution, discontinuous-Galerkin, and spectral-volume methods. The principle is based on a hyperbolic relaxation-system model for diffusion. It is to discretize the hyperbolic system by an advection scheme, and then derive a diffusion scheme by bringing it to the equilibrium limit in a rather simple manner. A distinguished feature of the proposed principle is that it automatically introduces a damping term into the derived diffusion scheme, which is essential for effective high-frequency error damping and, in some cases, for consistency also. Another useful feature is that the derived diffusion scheme has the same implementation structure as a corresponding advection scheme, which makes it remarkably simple to integrate it with the advection scheme for advection-diffusion problems. We demonstrate the general principle by constructing diffusion schemes on uniform grids in one dimension and unstructured grids in two dimensions, for node/cell-centered finite-volume, residual-distribution, discontinuous-Galerkin, and spectral-volume methods. Numerical results are presented to verify the accuracy of the diffusion schemes and to illustrate the importance of the damping term. It is also shown that derived diffusion schemes yield comparably or more accurate solutions than widely-used schemes for time-dependent diffusion problems on isotropic/anisotropic irregular triangular grids.

Contents

1	Introduction	3
2	First-Order Hyperbolic Diffusion System	4
3	General Principle	5
3.1	Diffusion Scheme from Advection Scheme	5
3.2	Relaxation Time	6
3.3	General Principle	7
4	One Dimension	7
4.1	Finite-Volume Diffusion Schemes	7
4.1.1	Formulation	7
4.1.2	Piecewise Constant Data	8
4.1.3	Reconstructed Piecewise Linear Data	8
4.1.4	Reconstructed Interfacial Linear Data	11
4.2	Residual-Distribution Diffusion Scheme	11
4.3	Discontinuous-Galerkin Diffusion Schemes	12
4.4	Spectral-Volume Diffusion Scheme	16
5	Two Dimensions	19
5.1	First-Order Hyperbolic Diffusion System	19
5.2	Relaxation Time	20

*Senior Research Scientist (hiro@nianet.org), National Institute of Aerospace, 100 Exploration Way, Hampton, VA 23666 USA
Copyright © 2010 by Hiroaki Nishikawa. Published by the American Institute of Aeronautics and Astronautics, Inc. with permission.

5.3	Node-Centered Finite-Volume Diffusion Schemes	20
5.3.1	Formulation	20
5.3.2	Edge-Based Diffusion Schemes	21
5.3.3	Node-Centered Compact Diffusion Schemes	24
5.3.4	Element-Based Compact Diffusion Schemes	24
5.4	Residual-Distribution Diffusion Schemes	26
5.4.1	LDA Diffusion Scheme	26
5.4.2	Lax-Wendroff Diffusion Scheme	29
5.5	Cell-Centered Finite-Volume Diffusion Schemes	30
5.5.1	Face-Midpoint Scheme	30
5.5.2	Edge-Midpoint Scheme	32
5.6	Discontinuous-Galerkin Diffusion Schemes	32
5.7	Spectral-Volume Diffusion Schemes	36
6	Numerical Results	39
6.1	One-Dimensional Problem	39
6.2	Two-Dimensional Problems	41
6.2.1	Isotropic Irregular Grids	41
6.2.2	Anisotropic Irregular Grids	45
7	Concluding Remarks	49
7.1	General Remarks	49
7.2	Derived Diffusion Schemes	49
7.3	Advection-Diffusion Problems	52
7.4	Future Work	53
Appendix A Bassi-Rebay and Penalized Bassi-Rebay Scheme		57
Appendix B Linearity-Preserving Finite-Volume Integration Formulas		59
B.1	Two Dimensions	59
B.1.1	Triangle	59
B.1.2	Quadrilateral - Median Dual	61
B.1.3	Quadrilateral - Alternative Dual	62
B.1.4	Triangular Grids	63
B.1.5	Quadrilateral Grids - Median Dual	64
B.1.6	Quadrilateral Grids - Alternative Dual	65
B.1.7	Mixed Triangular/Quadrilateral Grids	65
B.1.8	Linearity-Preserving Edge-Based Integration on General Mixed-Grids	67
B.2	Three Dimensions	67
B.2.1	Tetrahedron	67
B.2.2	Hexahedron	69
B.2.3	Prism	70
B.2.4	Pyramid	71
B.2.5	Hexahedral Grids and Tetrahedral Grids	72
B.2.6	Mixed Grids in Three Dimensions	72
Appendix C Average-Least-Squares Schemes		73
C.1	Edge-Normal Scheme	73
C.2	Face-Tangent Scheme	73
Appendix D Edge-Difference Form of the Braaten-Connell Diffusion scheme		74
Appendix E Inconsistency of the Galerkin Scheme under Positivity Enforcement		76
Acknowledgments		78
References		78

1. Introduction

*“So, how do you compute the gradient at the interface?”
“I don’t know. That’s not how I do it.”*

Towards highly efficient and accurate viscous simulations by Navier-Stokes codes, a great deal of effort has recently been devoted to the development of diffusion schemes with particular emphases on high-order methods [1, 2, 3, 4, 5, 6, 7, 8] and unstructured grid methods [9, 10, 11, 12, 13, 14, 15, 16]. A background approach of constructing diffusion schemes common to many works is to evaluate the solution gradient on a control volume boundary (e.g., by reconstruction) and compute the diffusive flux directly with them. The evaluation of the gradient is performed typically by using every nearby solution value around, reflecting, in a way, the isotropic nature of diffusion. Although seemingly flawless, this approach is known to result in unsatisfactory schemes: for example, non-convergent schemes [17, 18] and poor h -elliptic (poor high-frequency damping) schemes [19, 20, 21]. Hence, this approach by itself is not complete. In many methods, it is, therefore, augmented in one form or another in a manner specific to each discretization method. An example is the so-called penalty term in the discontinuous-Galerkin method [22, 23], and another is the edge-term in the finite-volume method [24, 25, 11]. These terms, playing a role of high-frequency damping, are known to improve the h -ellipticity and also to resolve the inconsistency problem that arises in high-order methods. However, these techniques are highly specialized. It is not straightforward to extend them to other methods, e.g., to the residual-distribution method for which a similar practical technique has not yet been developed. Even if a satisfactory diffusion scheme is devised, it may still encounter a compatibility problem (e.g., lost accuracy) when combined with an advection scheme for advection-diffusion problems [26, 27]. Although it may be possible to continue to devise another specialized fix for each problematic situation, what is really needed or highly desired is to improve the approach that is incomplete, so that practical diffusion schemes can be easily constructed for all discretization methods.

In high-order methods, several improved approaches have recently been proposed [1, 4, 7, 5, 6, 28, 29]. These approaches have been shown to generate some satisfactory schemes in the method they are proposed for. But it is, again, not immediately clear how to extend them to other discretization methods, particularly to the residual-distribution method. What is still missing, or highly sought, is a guiding principle that is independent of the discretization method, such as ‘upwinding’ for advection schemes. The failures of the common approach seem to be suggesting that the diffusion equation by itself is not enough to devise practical numerical schemes. Then, a possible avenue that we may take would be to explore other physical models for diffusion. A better model to be solved could then be found, but it is also possible to use such an alternative model just to derive a numerical scheme for the diffusion equation. The present paper demonstrates that the latter is indeed possible and it moreover gives birth to a general principle for deriving practical diffusion schemes for various discretization methods.

A model that we employ for deriving a diffusion scheme is a first-order hyperbolic relaxation system for diffusion [30, 31, 32], which we call the first-order hyperbolic diffusion system, or simply the hyperbolic diffusion system. The system being hyperbolic, a well-established principle for advection such as the upwinding principle is directly applicable. We propose to discretize this system (instead of the diffusion equation) by an upwind advection scheme, and derive a diffusion scheme from the result by discarding extra variables and their associated equations. As will be shown by many examples, the diffusion scheme thus derived will have two distinguished features. First, it is equipped with a damping term such as the penalty-term or the edge-term, which is inherited from the dissipation term of the generating advection scheme. The second feature is that the derived diffusion scheme is cast in the form of the corresponding advection scheme; thus it can be very efficiently and systematically integrated with an advection scheme for solving advection-diffusion problems. It may be pointed out at this point that the proposed principle is independent of the discretization method. In this paper, we demonstrate its general applicability by deriving diffusion schemes for node/cell-centered finite-volume, residual-distribution, discontinuous-Galerkin, and spectral-volume methods.

There exist other methods that utilize a first-order representation of the diffusion equation [4, 33], but their system is fundamentally different from the system we employ here. Our first-order system is hyperbolic in time while their system has no such characters. In deriving a diffusion scheme, we thus fully exploit the hyperbolic structure of the first-order system, enabling a straightforward design of a numerical scheme by a well-established principle for hyperbolic systems.

In Ref. [32], we employed the same first-order hyperbolic diffusion system to devise a new strategy for computing steady-state solutions of the diffusion equation: stable explicit time-stepping with $O(h)$ time step (instead of $O(h^2)$), where h is a mesh size, and simultaneous computation of the solution gradients with the equal order of accuracy to the main variable. We extended the strategy to the advection-diffusion equation

in Ref. [34] where we demonstrated a differential-level unification of advection and diffusion, and developed numerical schemes that were uniformly fast and accurate for all Reynolds numbers. These schemes are designed primarily for steady-state computations. In this paper, although using the same first-order system, we consider constructing ‘traditional’ time-accurate explicit diffusion schemes. Thus, the resulting schemes require no extra variables/equations to be stored/solved, and the explicit time step will be subject to the ‘traditional’ $O(h^2)$ restriction. A striking difference comes also in the definition of the relaxation time of the first-order hyperbolic diffusion system.

The paper is organized as follows. Section 2 describes the first-order hyperbolic diffusion system employed for deriving diffusion schemes. Section 3 presents the general principle for constructing diffusion schemes and the definition of the relaxation time suitable for deriving diffusion schemes. Section 4 describes applications of the proposed principle to the one-dimensional diffusion equation, deriving various finite-volume, residual-distribution, discontinuous-Galerkin, and spectral-volume schemes. Section 5 presents various diffusion schemes for the two-dimensional diffusion equation on unstructured grids: node/cell-centered finite-volume, residual-distribution, discontinuous-Galerkin, and spectral-volume schemes. Section 6 presents numerical results for time-dependent diffusion problems on uniform grids in one dimension and on isotropic/anisotropic irregular triangular grids in two dimensions. Finally, Section 7 contains conclusions and discussions on future developments.

2. First-Order Hyperbolic Diffusion System

Consider the diffusion equation:

$$u_t = \nu u_{xx}, \quad (2.1)$$

where $\nu > 0$. To derive a numerical scheme for the diffusion equation, we propose to discretize the first-order *hyperbolic diffusion system*:

$$\begin{aligned} u_t &= \nu p_x, \\ p_t &= (u_x - p)/T_r, \end{aligned} \quad (2.2)$$

that is asymptotically equivalent to the diffusion equation at large time, $t \gg T_r$ [30,31] when the extra variable p , called here the gradient variable, relaxes to u_x . This relaxation process is characterized by the time scale, T_r , which is called the relaxation time. In the vector form, the system (2.2) is written as

$$\mathbf{U}_t + \mathbf{F}_x = \mathbf{Q}, \quad (2.3)$$

where

$$\mathbf{U} = \begin{bmatrix} u \\ p \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} -\nu p \\ -u/T_r \end{bmatrix}, \quad \mathbf{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \begin{bmatrix} 0 & -\nu \\ -1/T_r & 0 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 0 \\ -p/T_r \end{bmatrix}. \quad (2.4)$$

The Jacobian matrix, \mathbf{A} , has a pair of real eigenvalues,

$$\pm \sqrt{\frac{\nu}{T_r}}, \quad (2.5)$$

and linearly independent eigenvectors [34]. Hence, the first-order system (2.2) is a hyperbolic system; various well-established techniques are available for discretization [35, 36, 37, 38]. The system describes two waves traveling in opposite directions at the same speed. Note, however, that as shown in Ref. [31], at large t compared to T_r , the waves will be damped out by a source term effect, recovering a smooth feature of the diffusion equation (2.1). This particular behavior is best illustrated by a Riemann problem for the hyperbolic diffusion system (2.2): two different states brought in contact begin to interact instantaneously. Typically, the initial discontinuity breaks up into two pieces carried by the two waves with a smooth Gaussian-type solution developed in between as shown in Figure 2.1. These waves eventually disappear by a damping effect of the source term, and the smooth solution spreads over the domain [31]. The initial stage is called the relaxation stage or frozen limit (the wave speeds (2.5) are called frozen speeds); thus, p is not equal to u_x yet. The last stage is called the equilibrium limit at which the waves have disappeared and we have $p = u_x$, thus satisfying the original diffusion equation (2.1). Further details can be found in Ref. [31]. Here, we are interested not to solve the hyperbolic diffusion system (2.2) but to derive time-accurate numerical schemes for the diffusion equation

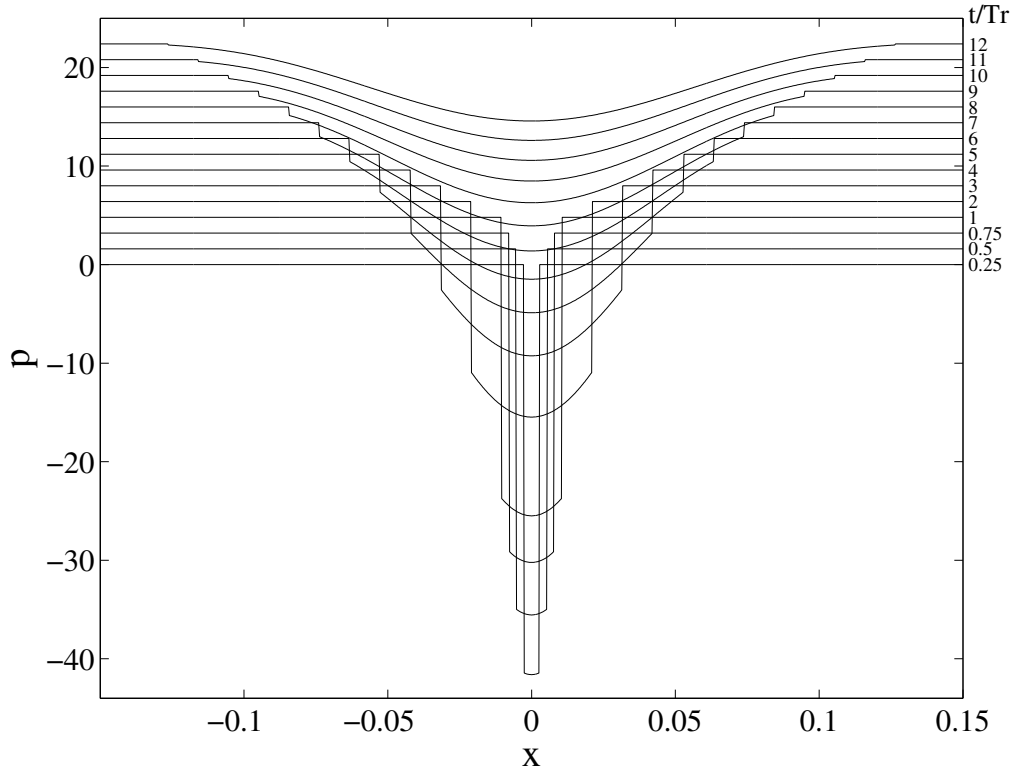


Figure 2.1. Exact solution to the Riemann problem for the gradient variable, p , computed by an analytical formula given in Ref. [31]. Plots are shown for various times: $t/T_r = 0.25, 0.5, 0.75, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12$, vertically shifted for better visibility. The initial condition is a jump in u and $p = 0$.

(2.1) by discretizing the hyperbolic diffusion system: derive an equilibrium scheme from a frozen scheme. Yet in other words, we derive a diffusion scheme from an advection scheme.

3. General Principle

3.1. Diffusion Scheme from Advection Scheme

For simplicity, but without loss of generality, consider a one-dimensional grid of uniform spacing, Δx . The numerical solution is stored at each data point j (a node or a cell center). The solution vector at a time level n is denoted by \mathbf{U}_j^n . Suppose we discretized the hyperbolic diffusion system (2.3) in the form:

$$\frac{\mathbf{U}_j^{n+1} - \mathbf{U}_j^n}{\Delta t} = -\mathbf{Res}_j^n + \mathbf{Q}_j^n, \quad (3.1)$$

where Δt is a time step, and $\mathbf{Res}_j^n = [Res_{j,1}^n, Res_{j,2}^n]^t$, where the superscript t denotes the transpose, is a discretized version of \mathbf{F}_x . We emphasize here that the system is hyperbolic and hence the type of scheme we need to define \mathbf{Res}_j^n is of advection, not diffusion. The specific form of the discretization is left open; it is not important here. Write out the scheme (3.1) by components:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -Res_{j,1}^n, \quad (3.2)$$

$$\frac{p_j^{n+1} - p_j^n}{\Delta t} = -Res_{j,2}^n - \frac{1}{T_r} p_j^n. \quad (3.3)$$

This scheme is not time-accurate for the diffusion equation (2.1) in general because p_j^n may not be an accurate approximation to the solution gradient. To make this scheme time-accurate, we must instantaneously drive the

gradient variable to the equilibrium limit: $p \rightarrow u_x$. We propose to achieve this by an explicit evaluation of p_j^n from the approximate solution of u (e.g., gradient reconstruction), so that the extra variable, p_j^n , no longer needs to be stored. Then, Equation (3.3) is also not needed any more, and we are left with a diffusion scheme:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -Res_{j,1}^n, \quad (3.4)$$

which is now time-accurate for the diffusion equation. We have thus derived a time-accurate diffusion scheme from an advection scheme applied to the hyperbolic diffusion system.

Several remarks are in order. It is expected that well-designed advection schemes incorporate some form of dissipation; it then enters the right hand side of Equation (3.4) as a damping term. This is how a damping term is introduced into the derived diffusion scheme. Also, it should be noted that the right hand side of Equation (3.4) retains the structure of the advection scheme from which it is derived. This feature will greatly simplify the integration of the diffusion scheme with a corresponding advection scheme for advection-diffusion equations. These key features will be illustrated later by many examples. It should be noted also that we have great flexibility in evaluating the gradient variable, p_j^n . Gradient reconstruction often requires an extended stencil, but a compact diffusion scheme can be constructed by evaluating the gradient with a compact stencil. Such examples will be shown later for a finite-volume method. We remark also that there are other ways to bring the advection scheme to the equilibrium limit. One is to take $T_r \rightarrow 0$; it will make the system extremely stiff. Another would be to determine p_j^n by forcing the right hand side of Equation (3.3) to vanish; it generally requires a solution of a global linear system (an exceptional case is discussed in Appendix A). The approach we took here may be considered as the simplest of all. Finally, we emphasize that although we begin with a system scheme for the hyperbolic diffusion system, the derived diffusion scheme is a scalar scheme for the diffusion equation. That is, the extra variable, p , is not stored; instead it is explicitly evaluated by the approximate solution of u .

3.2. Relaxation Time

To complete the construction of a diffusion scheme, it is necessary to define the relaxation time, T_r , which would enter the right hand side of Equation (3.4). Recall that we first discretize the hyperbolic diffusion system by an advection scheme. Hence, we should define T_r such that the hyperbolic diffusion system remains strongly hyperbolic. For explicit time-stepping schemes, it suffices to take T_r comparable to the maximum time step, so that the hyperbolic behavior is well kept during every time step. The maximum time step is defined based on the CFL condition:

$$\Delta t \leq \frac{\Delta x}{\sqrt{\nu/T_r}}, \quad (3.5)$$

where Δx is the minimum mesh spacing of a given grid. Then,

$$\Delta t_{\max} \equiv \frac{\Delta x}{\sqrt{\nu/T_r}} = \alpha T_r, \quad (3.6)$$

where α is a positive constant of $O(1)$ representing the ratio, $\Delta t_{\max}/T_r$. Solving Equation (3.6) for T_r , we obtain

$$T_r = \frac{\Delta x^2}{\alpha^2 \nu}. \quad (3.7)$$

We typically set $\alpha = 1$ for the physical reason mentioned above, but there are some other values having special properties as we will show later. Substituting Equation (3.7) into the CFL condition (3.5), we find

$$\Delta t \leq \frac{1}{\alpha} \frac{\Delta x^2}{\nu}. \quad (3.8)$$

This is the CFL condition for the derived diffusion scheme. Observe that a typical mesh-dependence of the time step for diffusion schemes has emerged: $\Delta t = O(\Delta x^2)$.

3.3. General Principle

A general principle for constructing diffusion schemes can be stated as follows:

Discretize the hyperbolic diffusion system by an advection scheme, then ignore the discrete equation for p , and instead approximate $p = u_x$ directly. The result is a time-accurate scheme for the diffusion equation, having a damping term inherited from the dissipation term of the advection scheme.

As discussed earlier and will be shown by many examples, the diffusion scheme derived from the principle will have, independently of the discretization method, two distinguished features:

1. It inherits the dissipation term of the advection scheme from which it is derived, which then acts as a high-frequency damping term.
2. It retains the structure of the advection scheme and can be implemented in the same way as the advection scheme.

The damping term, such as the edge-term in finite-volume methods or the penalty-term in discontinuous-Galerkin methods, is then automatically incorporated into the diffusion scheme by an appropriate choice of the advection scheme, e.g., the upwind scheme. This is a very useful feature. Simply following the principle, we obtain a diffusion scheme with a term responsible for high-frequency damping automatically built in. No special techniques nor extra considerations are necessary. Also, the second feature is expected to greatly simplify the integration of the diffusion scheme with an advection scheme for advection-diffusion problems (which typically requires two separate implementation frameworks). For example, in finite-volume methods, we can evaluate an advective flux and a diffusive flux at the same time by using a common pair of states at an interface, which brings a dramatic simplification in coding.

In the rest of the paper, we describe in details how the proposed principle can be applied to derive diffusion schemes in node/cell-centered finite-volume, residual-distribution, discontinuous-Galerkin, and spectral-volume methods.

4. One Dimension

4.1. Finite-Volume Diffusion Schemes

4.1.1. Formulation

Consider a one-dimensional grid of N cells with uniform spacing, Δx . The solution data are considered as cell-averages and stored at the cell center denoted by x_j , $j = 1, 2, 3, \dots, N$; thus, the data are discontinuous across cells. Integrating the hyperbolic diffusion system (2.3) over a cell, $I_j = [x_{j-1/2}, x_{j+1/2}]$, we obtain a standard semi-discrete finite-volume discretization:

$$\frac{d\mathbf{U}_j}{dt} = -\frac{1}{\Delta x} [\mathbf{F}_{j+1/2} - \mathbf{F}_{j-1/2}] + \frac{1}{\Delta x} \int_{I_j} \mathbf{Q} dx, \quad (4.1)$$

where $\mathbf{U}_j = [u_j, p_j]^t$ is the cell-averaged solution vector, and $\mathbf{F}_{j+1/2}$ is the interface flux to be defined. Note that the source term discretization is not important since it has a nonzero term only in the second equation which we will ignore. For time integration, we employ the forward-Euler time-stepping scheme for all schemes in this paper unless otherwise stated. The forward-Euler scheme is formally first-order accurate in time, but it gives second-order accuracy in space because the time step is proportional to Δx^2 for diffusion schemes as shown in Section 3.2. The interface flux, $\mathbf{F}_{j+1/2}$, can be determined, from two states, \mathbf{U}_L and \mathbf{U}_R , extrapolated from the left and right cells to the interface by any flux function suitable for the hyperbolic system. In this paper, we employ the upwind flux:

$$\begin{aligned} \mathbf{F}_{j+1/2} &= \frac{1}{2} [\mathbf{F}_R + \mathbf{F}_L] - \frac{1}{2} |\mathbf{A}| (\mathbf{U}_R - \mathbf{U}_L) \\ &= \frac{1}{2} [\mathbf{F}_R + \mathbf{F}_L] - \frac{1}{2} \sqrt{\frac{\nu}{T_r}} (\mathbf{U}_R - \mathbf{U}_L), \end{aligned} \quad (4.2)$$

which can be written by Equation (3.7) as

$$\mathbf{F}_{j+1/2} = \frac{1}{2} [\mathbf{F}_R + \mathbf{F}_L] - \frac{\nu\alpha}{2\Delta x} (\mathbf{U}_R - \mathbf{U}_L), \quad (4.3)$$

where \mathbf{F}_L and \mathbf{F}_R denote the physical flux evaluated by the left and right states respectively. In fact, due to the simplicity of the hyperbolic diffusion system, this flux is equivalent to the Rusanov flux [39], and if CFL = 1, it is also equivalent to the classical Lax-Friedrichs flux [40] and the Lax-Wendroff flux [41].

The discretization of the hyperbolic diffusion system (2.3) is now completed. At this point, the resulting finite-volume scheme (4.1) may not be time-accurate for the diffusion equation because p_j may not be an accurate approximation of u_x . To derive a time-accurate diffusion scheme, we discard the second component (i.e., the equation for dp_j/dt) and instead directly reconstruct p_j from u_j . Then, we are left with the first component:

$$\frac{du_j}{dt} = -\frac{1}{\Delta x} [f_{j+1/2} - f_{j-1/2}], \quad (4.4)$$

where

$$f_{j+1/2} = -\frac{\nu}{2} [p_R + p_L] - \frac{\nu\alpha}{2\Delta x} (u_R - u_L). \quad (4.5)$$

This is a time-accurate diffusion scheme. Note that the second term in the interface flux (4.5) is the damping term which comes directly from the dissipation term of the upwind scheme for the hyperbolic diffusion system. It is this term, as we will show later, that provides sufficient damping and can also improve the order of accuracy of the diffusion scheme. On the other hand, the first term approximates the physical flux; it is thus called the consistent term or part of the numerical flux. To complete the spatial discretization, we need to define the left and right states: u_L , u_R , p_L , and p_R . In particular, we need to evaluate the gradient variables, p_R and p_L , by using a set of discrete solutions of u (we thus do not store p_j). In doing so, the consistent term should always be retained and evaluated accurately as it is the term that makes the numerical flux consistent, thus resulting in a consistent scheme for the diffusion equation (2.1). It should be noted also that the derived diffusion scheme is implemented in the same way as the advection scheme: interface flux evaluated by two (possibly discontinuous) states. Examples that follow differ only in the definition of the left and right states.

4.1.2. Piecewise Constant Data

Assuming that the solution is piecewise constant, we immediately notice that the solution gradient is not available and thus the consistent part of the flux cannot be evaluated. Nevertheless, there is one special case where the scheme can be made consistent. We have at the interface,

$$p_L = p_R = 0, \quad (4.6)$$

$$u_L = u_j, \quad (4.7)$$

$$u_R = u_{j+1}. \quad (4.8)$$

$$(4.9)$$

Inserting these into the interface flux (4.5), we obtain

$$f_{j+1/2} = -\frac{\nu\alpha}{2\Delta x} (u_{j+1} - u_j). \quad (4.10)$$

Note that this flux has only the damping term since the consistent part has vanished identically. But this flux can be made to approximate the physical flux consistently, by setting

$$\alpha = 2. \quad (4.11)$$

Then, the resulting scheme is a standard three-point finite-difference scheme for diffusion:

$$\frac{du_j}{dt} = \frac{\nu}{\Delta x^2} (u_{j+1} - 2u_j + u_{j-1}). \quad (4.12)$$

In this case, there are no other choices for α to make the scheme consistent. As we will see in the next section, if we retain the consistent part of the flux, α will be a free parameter.

4.1.3. Reconstructed Piecewise Linear Data

We consider reconstructing the gradient in order to evaluate the consistent part of the flux function. Note that this would be the most natural option in our construction since it is a standard way to upgrade a first-order advection scheme to second-order [42]. Assuming that the solution is smooth for diffusion problems, we

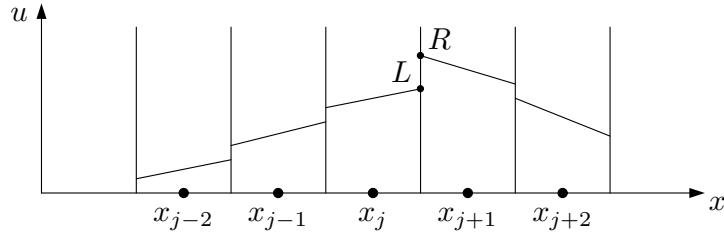


Figure 4.1. Discontinuous piecewise linear data in one dimension.

reconstruct the solution gradient within each cell (not at the interface) simply by the central-difference formula:

$$p_j = \frac{u_{j+1} - u_{j-1}}{2\Delta x}, \quad p_{j+1} = \frac{u_{j+2} - u_j}{2\Delta x}. \quad (4.13)$$

We now have a piecewise linear variation within each cell (see Figure 4.1), and the interface quantities can be evaluated as follows:

$$\begin{aligned} u_L &= u_j + \frac{1}{2} p_j \Delta x, & u_R &= u_{j+1} - \frac{1}{2} p_{j+1} \Delta x, \\ p_L &= p_j, & p_R &= p_{j+1}. \end{aligned} \quad (4.14)$$

These values are used to compute the interface flux (4.5), and the diffusion scheme is completely defined. Note that the resulting diffusion scheme is implemented in the same way as a second-order finite-volume advection scheme: interface flux computed by two discontinuous interface values. If such an advection scheme is implemented in a code already, it can be readily extended to an advection-diffusion scheme simply by adding the diffusive flux to the advective flux.

To investigate the accuracy and the damping property of the derived diffusion scheme, substitute (4.14) into the interface flux (4.5) to get

$$f_{j+1/2} = -\frac{\nu}{2} [p_j + p_{j+1}] - \frac{\nu\alpha}{2\Delta x} \left[u_{j+1} - u_j - \frac{1}{2} (p_j + p_{j+1}) \Delta x \right]. \quad (4.15)$$

We point out in passing that the damping term (the second term) is a quantity of $O(\Delta x^2)$ since $u_R - u_L = O(\Delta x^3)$ for smooth solutions and it is the consistent term (the first term) that approximates the solution gradient at the interface. Inserting this flux into Equation (4.4), we can write the derived diffusion scheme as

$$\frac{du_j}{dt} = \nu \frac{u_{j+2} - 2u_j + u_{j-2}}{4\Delta x^2} + \frac{\nu\alpha}{2\Delta x^2} \left[u_{j+1} - 2u_j + u_{j-1} - \frac{1}{4} (u_{j+2} - 2u_j + u_{j-2}) \right], \quad (4.16)$$

where the gradient variables, p_j and p_{j+1} , have been replaced by the central-difference formula (4.13). Note that this scheme has an extended five-point stencil (unless $\alpha = 2$ which leads to the three-point scheme). The scheme is consistent, not just for one but for many choices of α . To see this, insert a smooth function into the scheme and expand it to obtain

$$\frac{du_j}{dt} = \nu u_{xx} + \nu u_{xxxx} \left(\frac{1}{3} - \frac{\alpha}{8} \right) \Delta x^2 + O(\Delta x^4), \quad (4.17)$$

This shows that the scheme is consistent as well as second-order accurate for arbitrary α . It also follows that a particular choice,

$$\alpha = \frac{8}{3}, \quad (4.18)$$

makes the scheme fourth-order accurate. Now, insert a Fourier mode of phase angle $\beta \in [0, \pi]$, $u_0 \exp(i\beta x/\Delta x)$, where u_0 is a constant and $i = \sqrt{-1}$, into the scheme (4.16) to get

$$\frac{du_0}{dt} = \lambda u_0, \quad (4.19)$$

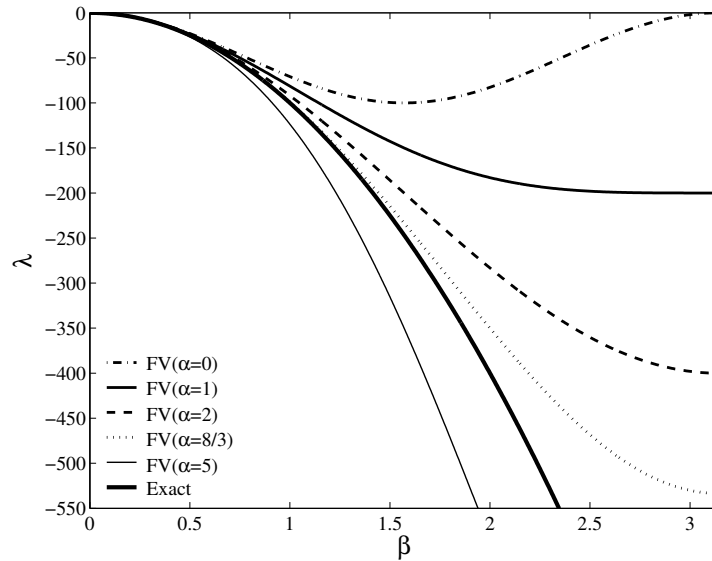


Figure 4.2. Damping factor λ for the exact operator, νu_{xx} and the finite-volume scheme with $\alpha = 0, 1, 2, 8/3,$ and 5 . $\nu = 1$ and $\Delta x = 0.1$.

where λ is the damping factor:

$$\lambda = \frac{1}{\Delta x^2} \left(-\nu \sin^2 \beta - 2\nu\alpha \sin^4 \frac{\beta}{2} \right). \quad (4.20)$$

It follows that the highest frequency error mode, $\beta = \pi$, cannot be damped if $\alpha = 0$, meaning that the consistent part does not provide sufficient damping. This is a typical odd-even decoupling problem which can be observed also in Equation (4.16). Then, the second term, which corresponds to the damping term, is the one that gives damping for high frequency modes. Figure 4.2 shows a plot of the damping factor for $\alpha = 0, 1, 2, 8/3,$ and 5 , compared with the damping factor of the exact diffusion operator. It confirms that the scheme with $\alpha = 0$ has poor high-frequency damping, and that increasing α leads to larger damping. In terms of the h -ellipticity, which is a measure of high-frequency damping representing the minimum damping factor over the high-frequency domain, $\beta = [\pi/2, \pi]$, (see Ref. [19] for details), the scheme with $\alpha = 0$ has zero h -ellipticity (it is therefore not h -elliptic), nonzero α makes the scheme h -elliptic, and increasing α improves the h -ellipticity of the scheme.

To estimate the maximum time step allowable for explicit time-stepping, we apply the forward-Euler time integration to obtain the stability condition:

$$\left| 1 + \frac{\Delta t}{\Delta x^2} \left(-\nu \sin^2 \beta - 2\nu\alpha \sin^4 \frac{\beta}{2} \right) \right| \leq 1. \quad (4.21)$$

This leads to the following time-step restriction:

$$\Delta t \leq \begin{cases} (2 - \alpha) \frac{\Delta x^2}{\nu}, & 0 \leq \alpha < 1, \\ \frac{1}{\alpha} \frac{\Delta x^2}{\nu}, & \alpha \geq 1. \end{cases} \quad (4.22)$$

As discussed in Section 3.2, we take $\alpha = 1$ to keep the hyperbolic diffusion system strongly hyperbolic over every time step. This choice may be considered as optimal in terms of accuracy and stability. Increasing α from 1 towards $8/3$ will lower the magnitude of the leading term of the truncation error, but reduce the stable time step. Increasing α beyond $8/3$ will increase the error and further reduce the stable time step. Taking α smaller than 1 increases the stable time step but increases the truncation error also. Therefore, $\alpha = 1$ is considered optimal for second-order calculations; or one may take $\alpha = 8/3$ if fourth-order accuracy is desired and the corresponding small time step is acceptable.

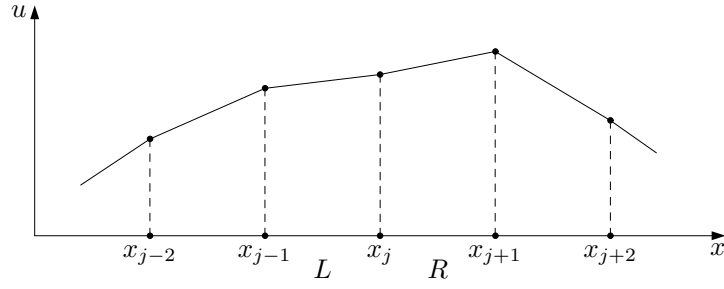


Figure 4.3. Continuous piecewise linear data in one dimension.

4.1.4. Reconstructed Interfacial Linear Data

Yet another possibility is to reconstruct the gradient at the interface,

$$p_{j+1/2} = \frac{u_{j+1} - u_j}{\Delta x}. \quad (4.23)$$

Using this common gradient in both adjacent cells to construct piecewise linear data, we obtain the interface quantities as

$$\begin{aligned} u_L &= u_j + \frac{1}{2} p_{j+1/2} \Delta x = \frac{u_j + u_{j+1}}{2}, & u_R &= u_{j+1} - \frac{1}{2} p_{j+1/2} \Delta x = \frac{u_j + u_{j+1}}{2}, \\ p_L &= p_{j+1/2}, & p_R &= p_{j+1/2}, \end{aligned} \quad (4.24)$$

i.e., the reconstructed data are continuous at the interface. Then, the interface flux (4.5) becomes

$$f_{j+1/2} = \frac{u_{j+1} - u_j}{\Delta x}. \quad (4.25)$$

Note that this flux consists of the consistent part only since the damping term has vanished identically. Using this flux, we will obtain the three-point scheme (4.12) again. Although we arrive at the same result, this time, we have consistency automatically (i.e., no need to specifically choose $\alpha = 2$). This is because we have accurately evaluated the consistent part of the flux. In effect, the above procedure corresponds to the lowest-order version of the recovery method [28] where the diffusive flux is evaluated directly by a continuous reconstruction across the interface. In higher dimensions, however, the damping term does not necessarily vanish even when a common gradient is used between two cells. It can still provide damping and improve the h -ellipticity as we will show later.

4.2. Residual-Distribution Diffusion Scheme

The proposed principle is directly applicable to the residual-distribution method [43]. In fact, the original idea of this paper was first discussed in Ref. [32] for residual-distribution schemes. In what follows, we briefly outline the construction of a residual-distribution diffusion scheme. It will turn out that the resulting scheme is identical to the finite-volume diffusion scheme derived in Section 4.1.1.

In the residual-distribution method, discrete solutions are viewed as point values and a cell is defined as an interval between two adjacent data points (see Figure 4.3). To discretize the hyperbolic diffusion system (2.3), we first define the cell-residual, Φ^C , by integrating Equation (2.3) over a cell, $C[x_j, x_{j+1}]$:

$$\Phi^C = \int_{x_j}^{x_{j+1}} (-\mathbf{A}\mathbf{U}_x + \mathbf{Q}) dx. \quad (4.26)$$

Assuming a piecewise linear variation of \mathbf{U} over the cell, we obtain

$$\Phi^C = -\mathbf{A}(\mathbf{U}_{j+1} - \mathbf{U}_j) + \bar{\mathbf{Q}}_C \Delta x, \quad (4.27)$$

where $\bar{\mathbf{Q}}_C = (\mathbf{Q}_{j+1} + \mathbf{Q}_j)/2$. We then distribute the cell-residual to the nodes by using a distribution matrix \mathcal{B}_j^C which gives fractions of Φ^C , Φ_j^C and Φ_{j+1}^C , to be distributed to the nodes, j and $j + 1$, respectively:

$$\Phi_j^C = \mathcal{B}_j^C \Phi^C, \quad \Phi_{j+1}^C = \mathcal{B}_{j+1}^C \Phi^C. \quad (4.28)$$

Here, we employ the following distribution matrix used in Ref. [32]:

$$\mathcal{B}_j^C = \frac{1}{2} \mathbf{I} - \frac{\tau}{2\Delta x} \mathbf{A}, \quad \mathcal{B}_{j+1}^C = \frac{1}{2} \mathbf{I} + \frac{\tau}{2\Delta x} \mathbf{A}, \quad \tau = \frac{\Delta x}{\sqrt{\nu/T_r}}. \quad (4.29)$$

This distribution scheme was derived as a Lax-Wendroff-type scheme, but it is also equivalent to the upwind scheme for the hyperbolic diffusion system in one dimension (see Ref. [32] for details). After completing the distribution step for all cells, we have the following semi-discrete equation at each node:

$$\frac{d\mathbf{U}_j}{dt} = \frac{1}{\Delta x} [\Phi_j^L + \Phi_j^R] = \frac{1}{\Delta x} [\mathcal{B}_j^L \Phi^L + \mathcal{B}_j^R \Phi^R], \quad (4.30)$$

where L and R denote the cells on the left and right of the node j , respectively (see Figure 4.3). The discretization is now completed; we then ignore the second component and obtain the residual-distribution diffusion scheme:

$$\frac{du_j}{dt} = \frac{1}{\Delta x} [\phi_j^L + \phi_j^R], \quad (4.31)$$

where

$$\phi_j^L = \frac{1}{2} \phi^L - \frac{\nu\alpha}{2} \left(\frac{\Delta u_L}{\Delta x} - \bar{p}_L \right), \quad \phi^L = \nu(p_j - p_{j-1}), \quad \Delta u_L = u_j - u_{j-1}, \quad \bar{p}_L = \frac{p_j + p_{j-1}}{2}, \quad (4.32)$$

$$\phi_j^R = \frac{1}{2} \phi^R + \frac{\nu\alpha}{2} \left(\frac{\Delta u_R}{\Delta x} - \bar{p}_R \right), \quad \phi^R = \nu(p_{j+1} - p_j), \quad \Delta u_R = u_{j+1} - u_j, \quad \bar{p}_R = \frac{p_{j+1} + p_j}{2}. \quad (4.33)$$

Here, the relation $\tau/T_r = \alpha$, which comes from Equation (3.7) and Equation (4.29), has been used to replace τ/T_r by α . Having lost the update equation for the discrete solution of p , we evaluate them by using the central-difference formula as we did for the finite-volume scheme in Section 4.1.3. This completes the construction of the diffusion scheme. We point out that the first term of the distributed residuals (e.g., $\frac{1}{2}\phi^L$) describes the isotropic distribution of the cell-residual. This appears a natural choice for diffusion, but it is known that this term alone cannot damp high-frequency error modes [21, 32]. It is, then again, the damping term (the second term) that gives sufficient damping. Note that the damping term sums up to zero over each cell; thus it does not affect the conservation: $\phi_j^C + \phi_{j+1}^C = \phi^C$.

As a residual-distribution scheme, the scheme is commonly implemented in the form presented above: calculations and distributions of the residuals over cells followed by updates of nodal solutions. But it can be implemented as a finite-difference or finite-volume scheme also. Expanding and rearranging, we write the diffusion scheme as

$$\frac{du_j}{dt} = \frac{\nu}{2\Delta x} (p_{j+1} - p_{j-1}) + \frac{\nu\alpha}{2\Delta x^2} \left[(u_{j+1} - 2u_j + u_{j-1}) - \frac{p_{j+1} - p_{j-1}}{2} \right]. \quad (4.34)$$

Then, replacing p_{j+1} and p_{j-1} by the central-difference formula (4.13), we obtain

$$\frac{du_j}{dt} = \nu \frac{u_{j+2} - 2u_j + u_{j-2}}{4\Delta x^2} + \frac{\nu\alpha}{2\Delta x^2} \left[u_{j+1} - 2u_j + u_{j-1} - \frac{1}{4}(u_{j+2} - 2u_j + u_{j-2}) \right]. \quad (4.35)$$

This is identical to the finite-volume scheme (4.16). Consequently, all analytical results in Section 4.1.3 equally apply to the residual-distribution diffusion scheme (4.31). The equivalence of the residual-distribution scheme and the finite-volume scheme in one dimension has long been known for hyperbolic systems. See Ref. [44] for example. We have just shown here that the equivalence is carried over to the diffusion equation.

4.3. Discontinuous-Galerkin Diffusion Schemes

Discontinuous Galerkin (DG) method is based on a weak formulation with discontinuous basis functions [17]. Multiplying the hyperbolic diffusion system (2.3) by a test function, v , and integrating over a computational cell, $I_j = [x_{j-1/2}, x_{j+1/2}]$, we obtain

$$\int_{I_j} v \mathbf{U}_t dx + \int_{I_j} v \mathbf{F}_x dx = \int_{I_j} v \mathbf{Q} dx. \quad (4.36)$$

Integration by parts gives

$$\int_{I_j} v \mathbf{U}_t dx = - [(v\mathbf{F})|_{j+1/2} - (v\mathbf{F})|_{j-1/2}] + \int_{I_j} v_x \mathbf{F} dx + \int_{I_j} v \mathbf{Q} dx. \quad (4.37)$$

To represent the solution, \mathbf{U} , discontinuous basis functions are employed that are defined independently within each cell. It consequently creates a gap at cell interfaces; the flux at each interface will be determined from the left and right values, \mathbf{U}_R and \mathbf{U}_L , by the upwind flux (4.3).

The P_0 method corresponds to piecewise constant data; it has been considered in 4.1. Here, we consider the P_1 method which corresponds to piecewise linear data:

$$\mathbf{U}(x) = \mathbf{U}_j + (x - x_j) \frac{\overline{\Delta \mathbf{U}}_j}{\Delta x}, \quad (4.38)$$

where $\mathbf{U}_j = [u_j, p_j]^t$ is the cell-averaged solution vector and $\overline{\Delta \mathbf{U}}_j = [\overline{\Delta u}_j, \overline{\Delta p}_j]^t$ is the undivided gradient vector. Choosing the basis functions, 1 and $(x - x_j)/\Delta x$, as test functions, we obtain the following evolution equations for \mathbf{U}_j and $\overline{\Delta \mathbf{U}}_j$:

$$\frac{d\mathbf{U}_j}{dt} = -\frac{1}{\Delta x} [\mathbf{F}_{j+1/2} - \mathbf{F}_{j-1/2}] + \frac{1}{\Delta x} \int_{I_j} \mathbf{Q} dx, \quad (4.39)$$

$$\frac{d(\overline{\Delta \mathbf{U}}_j)}{dt} = -\frac{6}{\Delta x} [\mathbf{F}_{j+1/2} + \mathbf{F}_{j-1/2}] + \frac{12}{\Delta x^2} \int_{I_j} \mathbf{F} dx + \frac{12}{\Delta x^2} \int_{I_j} \frac{x - x_j}{\Delta x} \mathbf{Q} dx, \quad (4.40)$$

where the interface flux is given by Equation (4.3). The volume integrals involving \mathbf{Q} are not important because \mathbf{Q} has a nonzero entry only in the second component which we will ignore. On the other hand, the volume integral of the flux in the second equation is important. An efficient formula, which reuses the interface fluxes, is

$$\int_{I_j} \mathbf{F} dx = \frac{\Delta x}{6} (\mathbf{F}_{j+1/2} + 4\mathbf{F}_j + \mathbf{F}_{j-1/2}), \quad (4.41)$$

where \mathbf{F}_j is evaluated directly by the solution at the cell-center. Now, discarding the equations for p_j and $\overline{\Delta p}_j$ from Equations (4.39) and (4.40), respectively, we obtain the following DG diffusion scheme:

$$\frac{du_j}{dt} = -\frac{1}{\Delta x} [f_{j+1/2} - f_{j-1/2}], \quad (4.42)$$

$$\begin{aligned} \frac{d(\overline{\Delta u}_j)}{dt} &= -\frac{6}{\Delta x} [f_{j+1/2} + f_{j-1/2}] + \frac{2}{\Delta x} (f_{j+1/2} + 4f_j + f_{j-1/2}) \\ &= -\frac{4}{\Delta x} [f_{j+1/2} - 2f_j + f_{j-1/2}], \end{aligned} \quad (4.43)$$

where the interface flux is given by Equation (4.5). Note that the right hand side of Equation (4.43) is a second-order accurate approximation of $-f_{xx} = -\nu u_{xxx}$. To evaluate the interface flux, e.g., at $j + 1/2$, we take the following values:

$$\begin{aligned} u_L &= u_j + \overline{\Delta u}_j/2, & u_R &= u_{j+1} - \overline{\Delta u}_{j+1}/2, \\ p_L &= \overline{\Delta u}_j/\Delta x, & p_R &= \overline{\Delta u}_{j+1}/\Delta x. \end{aligned} \quad (4.44)$$

We emphasize that at this point we have discarded both p_j and $\overline{\Delta p}_j$, and thus store only u_j and $\overline{\Delta u}_j$. Necessary quantities such as p_L and p_R are explicitly evaluated by the approximate solution of u as in Equation (4.44). The construction of a DG diffusion scheme is now completed. Note that the derived DG scheme is compact since the numerical flux involves only the left and right cells, and no extra variables nor equations are required. Some well-known DG diffusion schemes, e.g., the Bassi-Rebay scheme [33] or the Local DG (LDG) scheme [4], are also based on a first-order system representation of the diffusion equation, but their system involves neither time derivatives of auxiliary variables nor the relaxation time. Consequently, their system does not possess a hyperbolic character; the discretization is not trivial [4]. On the other hand, our first-order system is

hyperbolic in time; the discretization is straightforward since well-established techniques for hyperbolic systems are directly applicable. The DG diffusion scheme derived here is much similar to the interior-penalty schemes [22, 45, 46, 47, 48]. As we will show later, we rediscover some well-known interior-penalty schemes as variants of the diffusion scheme derived above. Note that the parameter α in our scheme is not mesh-dependent: the scheme is stable and second-order accurate with $\alpha = 1$ or any positive constant for arbitrary grids, and can be fourth-order accurate for a special value as shown below. It is of different nature from a kind of empiricism known for the classical interior penalty schemes [22].

It is instructive, although not necessary for implementation, to write the derived diffusion scheme in an expanded form. Inserting Equation (4.44), we can write the diffusive flux (4.5) as

$$f_{j+1/2} = -\frac{\nu}{2\Delta x} [\overline{\Delta u}_{j+1} + \overline{\Delta u}_j] - \frac{\nu\alpha}{2\Delta x} \left[u_{j+1} - u_j - \frac{1}{2} (\overline{\Delta u}_{j+1} + \overline{\Delta u}_j) \right], \quad (4.45)$$

which is then substituted into Equations (4.42) and (4.43) to get

$$\frac{du_j}{dt} = \frac{\nu}{\Delta x^2} \left[\frac{1}{2} \left(1 - \frac{\alpha}{2} \right) (\overline{\Delta u}_{j+1} - \overline{\Delta u}_{j-1}) + \frac{\alpha}{2} (u_{j+1} - 2u_j + u_{j-1}) \right], \quad (4.46)$$

$$\frac{d(\overline{\Delta u}_j)}{dt} = \frac{\nu}{\Delta x^2} \left[2(\overline{\Delta u}_{j+1} - 2\overline{\Delta u}_j + \overline{\Delta u}_{j-1}) + 4\alpha \left\{ \frac{u_{j+1} - u_{j-1}}{2} - \frac{\overline{\Delta u}_{j+1} + 2\overline{\Delta u}_j + \overline{\Delta u}_{j-1}}{4} \right\} \right]. \quad (4.47)$$

It is interesting that the two variables decouple if $\alpha = 2$, and the scheme reduces to the standard three-point scheme for u_j . To investigate stability and accuracy of the diffusion scheme, insert a Fourier mode,

$$\exp(i\beta x/\Delta x) \begin{pmatrix} u_0 \\ \overline{\Delta u}_0 \end{pmatrix} \quad (4.48)$$

into the scheme to get the evolution equation of the Fourier mode:

$$\frac{d}{dt} \begin{pmatrix} u_0 \\ \overline{\Delta u}_0 \end{pmatrix} = M(\beta) \begin{pmatrix} u_0 \\ \overline{\Delta u}_0 \end{pmatrix}, \quad (4.49)$$

where

$$M(\beta) = \frac{\nu e^{-i\beta}}{\Delta x^2} \begin{bmatrix} \frac{\alpha}{2} & \frac{\alpha-2}{4} \\ -2\alpha & 2-\alpha \end{bmatrix} + \frac{\nu}{\Delta x^2} \begin{bmatrix} -\alpha & 0 \\ 0 & -2(\alpha+2) \end{bmatrix} + \frac{\nu e^{i\beta}}{\Delta x^2} \begin{bmatrix} \frac{\alpha}{2} & -\frac{\alpha-2}{4} \\ 2\alpha & 2-\alpha \end{bmatrix}. \quad (4.50)$$

This matrix has the following eigenvalues:

$$\lambda_{1,2} = \frac{\nu}{\Delta x^2} \left[\frac{\alpha-4}{2} \cos \beta + \frac{3\alpha+4}{2} \pm \frac{1}{2} \sqrt{(\alpha-4)^2 (\cos^2 \beta + 1) + 2(\alpha+4)(3\alpha-4) \cos \beta + 8\alpha^2} \right], \quad (4.51)$$

where λ_1 and λ_2 are associated with the negative and positive signs, respectively. When $\alpha = 0$, they reduce to

$$\lambda_1 = 0, \quad \lambda_2 = \frac{4\nu}{\Delta x^2} (\cos \beta - 1), \quad (4.52)$$

Hence, the scheme has no damping for the error mode associated with λ_1 . Note also that the second eigenvalue is not consistent:

$$\lambda_2 = -\frac{2\nu\beta^2}{\Delta x^2} + O(\beta^4), \quad (4.53)$$

which differs from the exact operator, $-\nu\beta^2/\Delta x^2$, by a factor of 2. This inconsistency problem is well known [49]. An intuitive explanation for this failure would be that without the damping term, the diffusive flux (4.45) completely vanishes for a piecewise constant solution for which $\overline{\Delta u}_j = 0$; thus there is no way to remove a piecewise constant error mode. For nonzero α , expanding the eigenvalues for small β , we obtain

$$\lambda_1 = -\frac{\nu\beta^2}{\Delta x^2} + \frac{6-\alpha}{24\alpha} \frac{\nu\beta^4}{\Delta x^2} + O(\beta^6), \quad \lambda_2 = -\frac{4\alpha\nu}{\Delta x^2} + \left(\frac{\alpha}{2} - 1 \right) \frac{\nu\beta^2}{\Delta x^2} - \frac{(\alpha-2)(\alpha-3)}{24\alpha} \frac{\nu\beta^4}{\Delta x^2} + O(\beta^6), \quad (4.54)$$

The first mode approximates the exact diffusion operator with second-order accuracy. It can be fourth-order accurate if

$$\alpha = 6. \quad (4.55)$$

Increasing α , we can expect a large damping effect from the second eigenvalue λ_2 , but it would impose a severe restriction on the time step. For the forward-Euler time integration, we have the following stability conditions:

$$|1 + \lambda_1 \Delta t| \leq 1 \quad \text{and} \quad |1 + \lambda_2 \Delta t| \leq 1. \quad (4.56)$$

It can be shown that these conditions lead to the following restriction on the time step:

$$\Delta t \leq \begin{cases} \frac{1}{4} \frac{\Delta x^2}{\nu}, & 0 \leq \alpha < 2, \\ \frac{1}{2\alpha} \frac{\Delta x^2}{\nu}, & \alpha \geq 2. \end{cases} \quad (4.57)$$

Again, we take $\alpha = 1$ for the same physical reason as before. We may take $\alpha = 6$ if fourth-order accuracy is desired; this however severely limits the time step.

Various other DG diffusion schemes can be derived in the same framework, e.g., by different interface fluxes or quadrature formulas. Two examples are given below. It is beyond the scope of the present paper to study all possible DG diffusion schemes.

Example 1.:

The volume integral may be evaluated, instead of Equation (4.41), as follows:

$$\int_{I_j} f dx = \frac{\Delta x}{6} (f_{j+1/2} + 4\hat{f}_j + f_{j-1/2}), \quad (4.58)$$

$$\hat{f}_j = f_j + \frac{1}{8} (f_{j+1} - 2f_j + f_{j-1}). \quad (4.59)$$

This minor modification brings a change in the factor of the first term in Equation (4.47): 1 instead of 2. The modified matrix, $M'(\beta)$, has the following eigenvalues:

$$\lambda_1 = \frac{2\nu}{\Delta x^2} (\cos \beta - 1), \quad \lambda_2 = -\frac{\nu\alpha}{\Delta x^2} (\cos \beta + 3). \quad (4.60)$$

Observe that the first eigenvalue, λ_1 , corresponds to the eigenvalue of the standard three-point finite-difference scheme for diffusion, and that the second eigenvalue, λ_2 , provides damping for $\alpha > 0$. It is noted also that this scheme is consistent for all values of α although it loses damping ($\lambda_2 = 0$) for $\alpha = 0$. This example thus shows that a lack of damping does not necessarily imply inconsistency. As to accuracy, it is clear from the above eigenvalues that the scheme is second-order accurate for any value of α . Increasing α , we expect a large damping effect, but it would impose a severe restriction on the time step. It can be shown that the scheme is stable, with the forward-Euler time integration, under the following time-step restriction:

$$\Delta t \leq \begin{cases} \frac{1}{2} \frac{\Delta x^2}{\nu}, & 0 \leq \alpha < 1, \\ \frac{1}{2\alpha} \frac{\Delta x^2}{\nu}, & \alpha \geq 1. \end{cases} \quad (4.61)$$

Increasing α beyond 1 will severely limit the time step; $\alpha = 1$ may be a reasonable choice in terms of stability and damping. Note that this time step is twice as large as the previous one (4.57) for $0 \leq \alpha < 1$, still larger for $1 < \alpha < 2$, and become identical to Equation (4.57) for $\alpha \geq 2$.

Example 2.

The well-known (σ, μ) -schemes [23, 50] can be derived by yet another quadrature formula for the volume integral in Equation (4.40). Consider two possible evaluations for the volume integral:

$$I_1 = \int_{I_j} f dx = -\nu \int_{I_j} u_x dx = -\nu \int_{I_j} (u_j)_x dx = -\nu \overline{\Delta u_j}, \quad (4.62)$$

or

$$\begin{aligned}
I_2 &= \int_{I_j} f dx = -\nu \int_{I_j} u_x dx = -\nu [u|_{j+1/2} - u|_{j-1/2}] \\
&= -\nu \left[\frac{1}{2} (u_j^R + u_{j+1}^L) - \frac{1}{2} (u_{j-1}^R + u_j^L) \right] \\
&= -\nu \left[\frac{1}{2} (u_{j+1} - u_{j-1}) - \frac{1}{4} (\overline{\Delta u}_{j+1} - 2\overline{\Delta u}_j + \overline{\Delta u}_{j-1}) \right]. \tag{4.63}
\end{aligned}$$

We then take a weighted sum of the two:

$$\begin{aligned}
\int_{I_j} f dx &= (1-w) I_1 + w I_2 \\
&= -\nu \left[\frac{1}{2} w (u_{j+1} - u_{j-1}) - \frac{1}{4} (\overline{\Delta u}_{j+1} - 2\overline{\Delta u}_j + \overline{\Delta u}_{j-1}) \right. \\
&\quad \left. + \frac{1}{4} (1-w) (\overline{\Delta u}_{j+1} + 2\overline{\Delta u}_j + \overline{\Delta u}_{j-1}) \right], \tag{4.64}
\end{aligned}$$

where w may take any real value. This particular volume integral results in the following DG diffusion scheme:

$$\frac{du_j}{dt} = \frac{\nu}{\Delta x^2} \left[\frac{1}{2} \left(1 - \frac{\alpha}{2} \right) (\overline{\Delta u}_{j+1} - \overline{\Delta u}_{j-1}) + \frac{\alpha}{2} (u_{j+1} - 2u_j + u_{j-1}) \right], \tag{4.65}$$

$$\begin{aligned}
\frac{d(\overline{\Delta u}_j)}{dt} &= \frac{\nu}{\Delta x^2} \left[6 \left(\frac{\alpha}{2} - w \right) (u_{j+1} - u_{j-1}) + 3 (\overline{\Delta u}_{j+1} - 2\overline{\Delta u}_j + \overline{\Delta u}_{j-1}) \right. \\
&\quad \left. - 3 \left(\frac{\alpha}{2} - w \right) (\overline{\Delta u}_{j+1} + 2\overline{\Delta u}_j + \overline{\Delta u}_{j-1}) \right]. \tag{4.66}
\end{aligned}$$

This scheme is identified as the (σ, μ) -schemes with

$$\sigma = -w, \quad \mu = \frac{\alpha}{2}. \tag{4.67}$$

The so-called symmetric scheme [22] ($\sigma = 1$) corresponds to $w = -1$; the Baumann scheme [45, 46] ($\sigma = -1$) corresponds to $w = 1$. We have just learned that σ represents how the volume integral is evaluated and μ corresponds to the damping coefficient or equivalently half the ratio of the maximum explicit time step to the relaxation time ($\alpha = \Delta t_{\max}/T_r$).

4.4. Spectral-Volume Diffusion Scheme

In the spectral-volume method [18, 51], a cell, $I_j = [x_{j-1/2}, x_{j+1/2}]$, is systematically subdivided into a set of control volumes, $\{C_{j,i}\}$, $i = 1, 2, \dots, m$, where m depends on the accuracy sought. The cell thus divided is called the spectral volume (SV). The solution value is stored within each control volume as a volume-averaged value, $\mathbf{U}_{j,i} = [u_{j,i}, p_{j,i}]^t$. Given the volume-averaged solutions, a higher-order polynomial of order $m-1$ is then constructed within each spectral volume such that its volume average over a control volume is equal to $\mathbf{U}_{j,i}$. For second-order accuracy, a cell is divided into two control volumes ($m = 2$) defined by $C_{j,1} = [x_{j-1/2}, x_j]$ and $C_{j,2} = [x_j, x_{j+1/2}]$ (see Figure 4.4), and the solution values are defined by

$$\mathbf{U}_{j,1} = \frac{1}{\Delta x/2} \int_{x_{j-1/2}}^{x_j} \mathbf{U} dx, \quad \mathbf{U}_{j,2} = \frac{1}{\Delta x/2} \int_{x_j}^{x_{j+1/2}} \mathbf{U} dx. \tag{4.68}$$

We can then construct a polynomial of degree 1:

$$\mathbf{U}_j(x) = \mathbf{U}_{j,1} L_1(x) + \mathbf{U}_{j,2} L_2(x), \tag{4.69}$$

where

$$L_1(x) = -\frac{x - x_{j,2}}{\Delta x/2}, \quad L_2(x) = \frac{x - x_{j,1}}{\Delta x/2}. \tag{4.70}$$

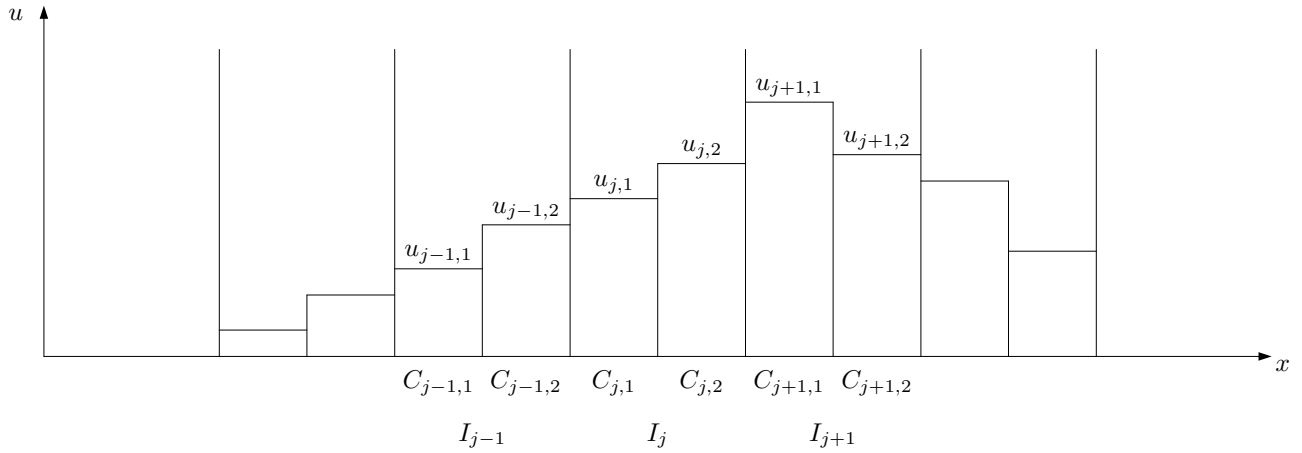


Figure 4.4. One-dimensional data for the second-order SV method.

The coordinates, $x_{j,1} = x_j - \Delta x/4$ and $x_{j,2} = x_j + \Delta x/4$, are the center coordinates of the control volumes. It can be shown by straightforward integration that the polynomial (4.69) satisfies

$$\frac{1}{\Delta x/2} \int_{C_{j,i}} \mathbf{U}_j(x) dx = \mathbf{U}_{j,i}, \quad i = 1, 2. \quad (4.71)$$

To derive a diffusion scheme, we follow the principle and begin by discretizing the hyperbolic diffusion system (2.3). Evolution equation for the volume-averaged values are obtained by integrating the system (2.3) over the control volumes:

$$\frac{d\mathbf{U}_{j,1}}{dt} = -\frac{1}{\Delta x/2} [\mathbf{F}_j - \mathbf{F}_{j-1/2}] + \frac{1}{\Delta x/2} \int_{C_{j,1}} \mathbf{Q} dx, \quad (4.72)$$

$$\frac{d\mathbf{U}_{j,2}}{dt} = -\frac{1}{\Delta x/2} [\mathbf{F}_{j+1/2} - \mathbf{F}_j] + \frac{1}{\Delta x/2} \int_{C_{j,2}} \mathbf{Q} dx, \quad (4.73)$$

where $\mathbf{F}_{j+1/2}$ and $\mathbf{F}_{j-1/2}$ are the SV-interface fluxes, and \mathbf{F}_j is the interior flux to be suitably defined. At the SV-interface where two independently defined polynomials meet, the solution can be discontinuous, and thus we employ the upwind flux:

$$\mathbf{F}_{j+1/2} = \frac{1}{2} [\mathbf{F}_R + \mathbf{F}_L] - \frac{\nu\alpha}{\Delta x} (\mathbf{U}_R - \mathbf{U}_L), \quad (4.74)$$

where the subscripts L and R denote the interior state and the neighboring state. Note that the dissipation coefficient of the above flux differs from that of the upwind flux (4.3) by factor 2 because each control volume here has a measure $\Delta x/2$, not Δx . On the other hand, the solution is continuous over the spectral volume, and therefore the interior flux is given by the physical flux at x_j : $\mathbf{F}_j = \mathbf{A}\mathbf{U}_j(x_j)$. Now, we derive a diffusion scheme by discarding the second component in each equation:

$$\frac{du_{j,1}}{dt} = -\frac{1}{\Delta x/2} [f_j - f_{j-1/2}], \quad (4.75)$$

$$\frac{du_{j,2}}{dt} = -\frac{1}{\Delta x/2} [f_{j+1/2} - f_j], \quad (4.76)$$

where

$$f_{j+1/2} = -\frac{\nu}{2} [p_R + p_L] - \frac{\nu\alpha}{\Delta x} (u_R - u_L), \quad (4.77)$$

$$f_j = -\nu \frac{du_j(x)}{dx}. \quad (4.78)$$

We complete the derivation by defining the left and right states:

$$\begin{aligned} u_L &= u_j(x_{j+1/2}), & u_R &= u_{j+1}(x_{j+1/2}), \\ p_L &= \frac{du_j(x)}{dx}, & p_R &= \frac{du_{j+1}(x)}{dx}. \end{aligned} \quad (4.79)$$

Again, we set $\alpha = 1$ but there is a special value that makes the scheme fourth-order accurate as shown below. Note that we have discarded both $p_{j,1}$ and $p_{j,2}$, and thus store only $u_{j,1}$ and $u_{j,2}$. Necessary quantities, such as p_L and p_R , are evaluated by differentiating the polynomial representation of the solution in the corresponding spectral volume. The derived SV diffusion scheme is compact: it involves the neighboring spectral-volumes only. The scheme is very similar to the penalty scheme considered in Ref. [18].

To examine accuracy and stability of the derived SV diffusion scheme, insert into the system, (4.75) and (4.76), a Fourier mode,

$$\exp(i\beta x/\Delta x) \begin{pmatrix} u_{0,1} \\ u_{0,2} \end{pmatrix}, \quad (4.80)$$

where $u_{0,1}$ and $u_{0,2}$ are constant amplitudes, to get

$$\frac{d}{dt} \begin{pmatrix} u_{0,1} \\ u_{0,2} \end{pmatrix} = M_{sv}(\beta) \begin{pmatrix} u_{0,1} \\ u_{0,2} \end{pmatrix}. \quad (4.81)$$

The matrix, $M_{sv}(\beta)$, is given by

$$M_{sv}(\beta) = \frac{\nu e^{-i\beta}}{\Delta x^2} \begin{bmatrix} 2 - \alpha & 3\alpha - 2 \\ 0 & 0 \end{bmatrix} + \frac{\nu}{\Delta x^2} \begin{bmatrix} -(3\alpha + 2) & \alpha + 2 \\ \alpha + 2 & -(3\alpha + 2) \end{bmatrix} + \frac{\nu e^{i\beta}}{\Delta x^2} \begin{bmatrix} 0 & 0 \\ 3\alpha - 2 & 2 - \alpha \end{bmatrix}. \quad (4.82)$$

This matrix has the following eigenvalues:

$$\lambda_{1,2} = \frac{\nu}{\Delta x^2} \left[(2 - \alpha) \cos \beta - (3\alpha + 2) \pm \sqrt{(\alpha - 2)^2 \cos^2 \beta + 2(3\alpha - 2)(\alpha + 2) \cos \beta + (3\alpha + 2)^2 - 16\alpha} \right], \quad (4.83)$$

where λ_1 and λ_2 are associated with the positive and negative signs, respectively. When $\alpha = 0$, they reduce to

$$0, \quad \frac{4\nu}{\Delta x^2}(\cos \beta - 1). \quad (4.84)$$

These are exactly the same as those of the DG scheme with $\alpha = 0$, and thus the scheme is inconsistent. For nonzero α , we can expand them for small β to get

$$\lambda_1 = -\frac{\nu\beta^2}{\Delta x^2} + \frac{3 - \alpha}{24\alpha} \frac{\nu\beta^4}{\Delta x^2} + O(\beta^6), \quad (4.85)$$

$$\lambda_2 = -\frac{8\alpha\nu}{\Delta x^2} + \frac{(\alpha - 1)\nu\beta^2}{\Delta x^2} - \frac{(\alpha - 1)(2\alpha - 3)}{24\alpha} \frac{\nu\beta^4}{\Delta x^2} + O(\beta^6). \quad (4.86)$$

It follows that the scheme is stable and second-order accurate for any non-negative α except for $\alpha = 3$ which makes the scheme fourth-order accurate. Note, as pointed out in Ref. [51], that the predicted accuracy is obtained in $u_j(x)$, not necessarily in the control-volume averages, $u_{j,1}$ and $u_{j,2}$, individually. Numerical experiments show that the control-volume averages are generally second-order accurate but only third-order accurate for $\alpha = 3$. For the forward-Euler time-stepping scheme, we obtain the following stability condition:

$$\Delta t \leq \begin{cases} \frac{1}{4} \frac{\Delta x^2}{\nu}, & 0 \leq \alpha < 1, \\ \frac{1}{4\alpha} \frac{\Delta x^2}{\nu}, & \alpha \geq 1. \end{cases} \quad (4.87)$$

This condition is essentially the same as the condition (4.57) for the DG scheme if the factor 2 in the control volume is taken into account. The same is true, in fact, for the eigenvalues. Note, however, that the discrete equations are different, and thus the numerical solutions will be different. See Ref. [51] for more details on the similarities and differences of the SV and DG methods.

5. Two Dimensions

5.1. First-Order Hyperbolic Diffusion System

We consider constructing numerical schemes for the diffusion equation in two dimensions:

$$u_t = \nu(u_{xx} + u_{yy}). \quad (5.1)$$

The first-order hyperbolic diffusion system is given by

$$\begin{aligned} u_t &= \nu(p_x + q_y), \\ p_t &= (u_x - p)/T_r, \\ q_t &= (u_y - q)/T_r, \end{aligned} \quad (5.2)$$

where p and q are the gradient variables which relax to the solution derivatives, u_x and u_y , respectively. Write the system in the vector form,

$$\mathbf{U}_t + \mathbf{F}_x + \mathbf{G}_y = \mathbf{Q}, \quad (5.3)$$

where

$$\mathbf{U} = \begin{bmatrix} u \\ p \\ q \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} -\nu p \\ -u/T_r \\ 0 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} -\nu q \\ 0 \\ -u/T_r \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 0 \\ -p/T_r \\ -q/T_r \end{bmatrix}. \quad (5.4)$$

The flux Jacobians are given by

$$\mathbf{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \begin{bmatrix} 0 & -\nu & 0 \\ -1/T_r & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \frac{\partial \mathbf{G}}{\partial \mathbf{U}} = \begin{bmatrix} 0 & 0 & -\nu \\ 0 & 0 & 0 \\ -1/T_r & 0 & 0 \end{bmatrix}. \quad (5.5)$$

Consider a projected Jacobian along an arbitrary vector, $\mathbf{n} = (n_x, n_y)$:

$$\mathbf{A}_n = \mathbf{A}n_x + \mathbf{B}n_y = \begin{bmatrix} 0 & -\nu n_x & -\nu n_y \\ -n_x/T_r & 0 & 0 \\ -n_y/T_r & 0 & 0 \end{bmatrix}. \quad (5.6)$$

The projected Jacobian has the following eigenvalues:

$$\lambda_1 = -\sqrt{\frac{\nu}{T_r}}, \quad \lambda_2 = \sqrt{\frac{\nu}{T_r}}, \quad \lambda_3 = 0. \quad (5.7)$$

The first two eigenvalues are identical to those in one dimension. Note also that these eigenvalues are independent of \mathbf{n} , and therefore the system describes a wave propagating isotropically. The third eigenvalue corresponds to the inconsistency damping mode [32], acting on the components of p and q such that $q_x - p_y \neq 0$; it is irrelevant in the present framework. The absolute Jacobian, $|\mathbf{A}_n|$, which is needed to define the upwind flux, is constructed by the right-eigenvector matrix, \mathbf{R}_n , and the diagonal eigenvalue-matrix, $\mathbf{\Lambda}_n$,

$$\mathbf{R}_n = \begin{bmatrix} \sqrt{\nu T_r} & -\sqrt{\nu T_r} & 0 \\ n_x & n_x & -n_y \\ n_y & n_y & n_x \end{bmatrix}, \quad \mathbf{\Lambda}_n = \begin{bmatrix} -\sqrt{\nu/T_r} & 0 & 0 \\ 0 & \sqrt{\nu/T_r} & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (5.8)$$

as follows:

$$|\mathbf{A}_n| = \mathbf{R}_n \mathbf{\Lambda}_n \mathbf{R}_n^{-1} = \sqrt{\frac{\nu}{T_r}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & n_x^2 & n_x n_y \\ 0 & n_x n_y & n_y^2 \end{bmatrix}. \quad (5.9)$$

It is the first row that is relevant to diffusion schemes.

5.2. Relaxation Time

The relaxation time, T_r , can be derived as in Section 3.2 based on the maximum explicit time-step. The precise form of T_r may depend on the mesh type and/or the discretization method, but it can be expressed generally in the form,

$$T_r = \frac{L_r^2}{\alpha^2 \nu}, \quad (5.10)$$

where L_r is a length scale that must be suitably defined for a given discretization, and α is again a positive constant representing the ratio of the maximum explicit time step to the relaxation time. As in one dimension, we typically set $\alpha = 1$ to keep the system strongly hyperbolic over every time step. It is important to note in two dimensions, however, that a length scale arising in the CFL condition is typically not a mesh size, h , but $h/2$. This is well known to be the case for donor-cell-type schemes, i.e., those involving only the face-adjacent neighbors in the stencil [35,38]. For example, if we apply the upwind difference scheme defined in each coordinate direction, the time step is restricted based on the sum of the CFL number (the ratio of the physical and numerical wave speeds) in each direction with spacing Δx and Δy respectively:

$$\frac{\Delta t}{\Delta x} \sqrt{\frac{\nu}{T_r}} + \frac{\Delta t}{\Delta y} \sqrt{\frac{\nu}{T_r}} \leq 1. \quad (5.11)$$

For a uniform mesh with $\Delta x = \Delta y = h$, this condition results in

$$\Delta t \leq \frac{h/2}{\sqrt{\nu/T_r}}. \quad (5.12)$$

Then, requiring the maximum time-step to be comparable to the relaxation time as in Section 3.2, we obtain

$$T_r = \frac{h^2}{4\alpha^2 \nu}, \quad (5.13)$$

which corresponds to the formula (5.10) with

$$L_r = \frac{h}{2}. \quad (5.14)$$

Based on this observation and for simplicity, we define L_r as half a typical mesh spacing for two-dimensional schemes considered in this paper. In three dimensions, for similar reasons, L_r may be defined as the third of a typical mesh spacing.

It should be noted also that the length scale should be defined in the direction relevant to the wave propagation. For example, in finite-volume methods, L_r is taken to be a length measured in the direction of the face normal along which a Riemann problem is considered. As will be shown, this definition brings a significant distinction of two-dimensional (as well as three-dimensional) schemes from one-dimensional schemes. It introduces a measure of grid-skewness into the damping coefficient in such a way that it increases damping for highly-skewed grids, enabling robust and accurate computations on typical viscous grids.

5.3. Node-Centered Finite-Volume Diffusion Schemes

5.3.1. Formulation

We begin by dividing a domain of interest into a set of elements, $\{E\}$, including both triangles and quadrilaterals (see Figure 5.1). We denote the associated set of nodes by $\{J\}$. In node-centered schemes, we store the solution values at nodes, and approximate the integral of the hyperbolic diffusion system over a dual control volume, Ω_j , around a node $j \in \{J\}$:

$$\frac{d\mathbf{U}_j}{dt} = \frac{1}{V_j} \int_{\Omega_j} (-\mathbf{F}_x - \mathbf{G}_y + \mathbf{Q}) dV \quad (5.15)$$

$$= -\frac{1}{V_j} \oint_{\partial\Omega_j} \mathbf{H} dA + \int_{\Omega_j} \mathbf{Q} dV, \quad (5.16)$$

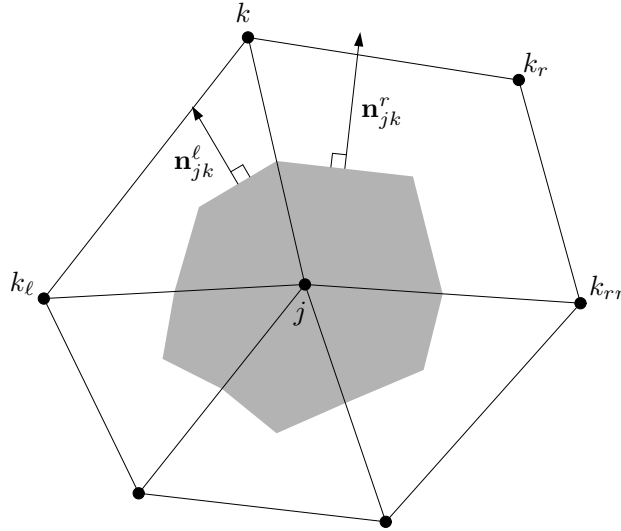


Figure 5.1. Dual control volume for node-centered finite-volume schemes with unit normals associated with an edge, $\{j, k\}$.

where V_j denotes the volume of the dual control volume, dV denotes the infinitesimal volume, $\partial\Omega_j$ denotes the boundary of the dual control volume, dA is the infinitesimal area of the boundary, and \mathbf{H} is the flux projected along the unit outward normal vector of the boundary, $\hat{\mathbf{n}}$:

$$\mathbf{H} = [\mathbf{F}, \mathbf{G}] \cdot \hat{\mathbf{n}}. \quad (5.17)$$

Focusing on the boundary integral in Equation (5.16), we consider three different discretization methods: edge-based method, node-centered compact method, and element-based method.

5.3.2. Edge-Based Diffusion Schemes

A common edge-based finite-volume scheme is based on the following edge-based quadrature formula, approximating Equation (5.16):

$$\frac{d\mathbf{U}_j}{dt} = -\frac{1}{V_j} \sum_{k \in \{K_j\}} \Phi_{jk} A_{jk} + \frac{1}{V_j} \int_{\Omega_j} \mathbf{Q} dV, \quad (5.18)$$

where $\{K_j\}$ is a set of neighbors of j , Φ_{jk} is a numerical flux along the directed area vector (see Figure 5.1),

$$\mathbf{n}_{jk} = \mathbf{n}_{jk}^l + \mathbf{n}_{jk}^r, \quad (5.19)$$

and A_{jk} is the magnitude of the directed area vector, i.e., $A_{jk} = |\mathbf{n}_{jk}|$. We remark that this edge-based quadrature formula is exact for linear fluxes only for triangles or parallelograms; there may be errors committed along an edge shared by a triangle and a quadrilateral [20, 52]; yet another care must be taken along boundary edges [53] (see Appendix B for more details). For the numerical flux, we employ the upwind flux:

$$\Phi_{jk} = \frac{1}{2} [\mathbf{H}_{jk}(\mathbf{U}_R) + \mathbf{H}_{jk}(\mathbf{U}_L)] - \frac{1}{2} |\mathbf{A}_n| (\mathbf{U}_R - \mathbf{U}_L), \quad (5.20)$$

where \mathbf{U}_R and \mathbf{U}_L are extrapolated solution vectors at the midpoint of the edge, $\{j, k\}$, the absolute Jacobian, $|\mathbf{A}_n|$, is defined based on the directed area vector, and \mathbf{H}_{jk} is the physical flux projected along the directed area vector:

$$\mathbf{H}_{jk} = [\mathbf{F}, \mathbf{G}] \cdot \hat{\mathbf{n}}_{jk}, \quad \hat{\mathbf{n}}_{jk} = \frac{\mathbf{n}_{jk}}{A_{jk}}. \quad (5.21)$$

The semi-discrete equation (5.18) is then integrated in time by the forward-Euler time-stepping scheme. The global time step, Δt , is defined as the minimum of the local time step, Δt_j , restricted by the local CFL condition:

$$\Delta t_j \leq \frac{2V_j}{\sum_{k \in \{K_j\}} \sqrt{\frac{\nu}{T_r}} A_{jk}}. \quad (5.22)$$

Having completed the discretization of the hyperbolic diffusion system, we now ignore the second and third equations in Equation (5.18) and obtain the following edge-based diffusion scheme:

$$\frac{du_j}{dt} = -\frac{1}{V_j} \sum_{k \in \{K_j\}} \phi_{jk} A_{jk}, \quad (5.23)$$

where

$$\phi_{jk} = -\frac{\nu}{2} [(p, q)_R + (p, q)_L] \cdot \hat{\mathbf{n}}_{jk} - \frac{1}{2} \sqrt{\frac{\nu}{T_r}} (u_R - u_L). \quad (5.24)$$

Note that the second term is the damping term inherited from the dissipation term of the upwind scheme. As will be shown later, this term involves the so-called edge-term which is considered as important for high-frequency damping. The relaxation time, T_r , is defined as in Equation (5.10), thus giving

$$\phi_{jk} = -\frac{\nu}{2} [(p, q)_R + (p, q)_L] \cdot \hat{\mathbf{n}}_{jk} - \frac{\nu\alpha}{2L_r} (u_R - u_L), \quad (5.25)$$

where, again, we set $\alpha = 1$, but there are special values as will be discussed later. For unstructured grids, the length scale L_r must be defined at each edge, so that the same interface flux is used per edge and the conservation is not violated. Based on the fact that L_r defines the wave speed of the hyperbolic diffusion system in a Riemann problem in the direction $\hat{\mathbf{n}}_{jk}$, it would be reasonable to take the edge length projected along $\hat{\mathbf{n}}_{jk}$:

$$L_r = \frac{1}{2} |\Delta \mathbf{l}_{jk} \cdot \hat{\mathbf{n}}_{jk}|, \quad (5.26)$$

where $\Delta \mathbf{l}_{jk} = (x_k - x_j, y_k - y_j)$. To evaluate the interface flux, the left and right states must be defined. One way is to reconstruct the solution gradient at every node and extrapolate the solution to the interface from each end node, just like a reconstruction-based advection scheme. The gradient at a node, j , can be computed, for example, by the Green-Gauss formula applied over a set $\{E_j\}$ of elements that share the node j :

$$(\nabla u)_j = \frac{\sum_{E \in \{E_j\}} (\nabla u)^E V^E}{\sum_{E \in \{E_j\}} V^E}, \quad (5.27)$$

where V^E is the volume of the element E and $(\nabla u)^E$ is the Green-Gauss gradient over that element. This formula is exact for linear solutions and can be implemented also in the edge-based form under the same limitations mentioned earlier for the edge-based quadrature formula (see Appendix B for details). Alternatively, the gradient may be computed by least-squares type methods. Once the solution gradients are made available at all nodes, the interface values can be obtained as follows:

$$\begin{aligned} u_L &= u_j + \frac{1}{2} (\nabla u)_j \cdot \Delta \mathbf{l}_{jk}, & u_R &= u_k - \frac{1}{2} (\nabla u)_k \cdot \Delta \mathbf{l}_{jk}, \\ (p, q)_L &= (\nabla u)_j, & (p, q)_R &= (\nabla u)_k. \end{aligned} \quad (5.28)$$

The diffusion scheme is now completely defined. We point out that the derived diffusion scheme, Equation (5.23) with the flux (5.25), is a general edge-based diffusion scheme directly applicable to arbitrary grids. Extension to three dimensions is straightforward. We remark also that this edge-based diffusion scheme can be implemented in the same framework of advection schemes: reconstruction, extrapolation, and interface flux evaluation. If such mechanisms are already in place, the diffusion scheme can be directly implemented simply by adding the interface flux. For solving advection-diffusion type equations, this will make the code very simple and efficient. Note also that the scheme reduces to a standard five-point finite-difference formula on a uniform quadrilateral mesh. In that case, the scheme is essentially equivalent to the one-dimensional scheme derived in Section 4.1.3 in each coordinate direction, and thus can be made fourth-order accurate by $\alpha = 4/3$ as shown Section 4.1.3 (not $\alpha = 8/3$ because of the difference in the factor $1/2$ in the definition of L_r). As will be shown later, even on irregular grids, the scheme with $\alpha = 4/3$ produces, although not fourth-order accurate, much more accurate solutions than other node-centered schemes.

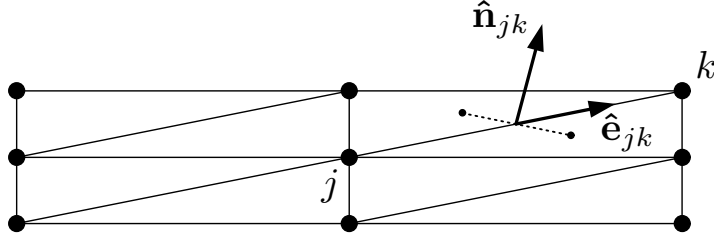


Figure 5.2. Example of skewed grids. $\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk} \rightarrow 0$ as the vertical spacing decreases.

To gain more insight, we insert the left and right states into the diffusive flux (5.25) to get

$$\phi_{jk} = -\frac{\nu}{2} [(\nabla u)_k + (\nabla u)_j] \cdot \hat{\mathbf{n}}_{jk} - \frac{\nu\alpha}{|\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}|} \left[\frac{u_k - u_j}{\Delta l_{jk}} - \frac{1}{2} \{(\nabla u)_k + (\nabla u)_j\} \cdot \hat{\mathbf{e}}_{jk} \right], \quad (5.29)$$

where $\Delta l_{jk} = |\Delta \mathbf{l}_{jk}|$ and $\hat{\mathbf{e}}_{jk} = \Delta \mathbf{l}_{jk} / |\Delta \mathbf{l}_{jk}|$. It shows that the so-called edge-term, $\frac{u_k - u_j}{\Delta l_{jk}}$, has been incorporated into the diffusion scheme. It has been automatically introduced by going through the advection scheme; no specific technique nor consideration was necessary to include it. Note also that as in one dimension, it is the consistent part (the first term) that approximates the solution gradient at the interface; the damping term (the second term) is a quantity of $O(h^m)$ for m -th order accurate gradients. Now, there is a special choice for α :

$$\alpha = (\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}) |\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}|. \quad (5.30)$$

The diffusive flux (5.29) then becomes

$$\phi_{jk} = -\nu \left[\frac{u_k - u_j}{\Delta l_{jk}} \hat{\mathbf{e}}_{jk} + \overline{\nabla u} - (\overline{\nabla u} \cdot \hat{\mathbf{e}}_{jk}) \hat{\mathbf{e}}_{jk} \right] \cdot \hat{\mathbf{n}}_{jk}, \quad (5.31)$$

where

$$\overline{\nabla u} = \frac{1}{2} [(\nabla u)_k + (\nabla u)_j]. \quad (5.32)$$

This flux corresponds to one of the widely-used average-least-squares schemes [11, 25, 54], often called the edge-normal average-least-squares (Avg-LSQ-EN) scheme. The Avg-LSQ-EN scheme is constructed such that the gradient in the diffusive flux (the term in the square bracket in Equation (5.31)) recovers the edge-term along $\hat{\mathbf{e}}_{jk}$ and the average-least-squares gradient, $\overline{\nabla u}$, along the direction normal to $\hat{\mathbf{e}}_{jk}$ (see Appendix C). The edge-term is thus deliberately incorporated to improve the h -ellipticity. There is another special choice [55]:

$$\alpha = \frac{|\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}|}{\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}}. \quad (5.33)$$

This results in the so-called face-tangent average-least-squares (Avg-LSQ-FT) scheme [56]. The Avg-LSQ-FT scheme is constructed such that the gradient in the diffusive flux recovers the edge-term along $\hat{\mathbf{e}}_{jk}$ and the average-least-squares gradient along the direction tangential to the face (see Ref. [56] and also Appendix C). Comparing these schemes in the form (5.29), we see that the damping coefficient generally depends on the inner-product, $\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}$, which may be considered as a skewness measure. For the Avg-LSQ-EN scheme, the damping coefficient is proportional to $\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}$. For the Avg-LSQ-FT scheme and the derived scheme (5.29) with any positive real value of α , it is inversely proportional to $\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}$. Consequently, the Avg-LSQ-EN scheme will suffer from a lack of damping for highly-skewed grids (typical viscous grids such as shown in Figure 5.2) for which $\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk} \rightarrow 0$. On the other hand, the Avg-LSQ-FT scheme and the derived scheme will gain a large damping effect. On such grids, the Avg-LSQ-FT scheme is in fact very similar to the derived scheme with $\alpha = 1$; they differ only in the sign of the damping coefficient. If the skewness angle exceeds 90° , the damping coefficient of the Avg-LSQ-FT scheme goes negative (damping turns into amplifying), while that of the derived diffusion scheme with positive α remains positive. For a grid with no skewness ($\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk} = 1$), such as a structured quadrilateral grid, both average-least-squares schemes will become identical to the derived scheme with $\alpha = 1$, and all reduce to a standard five-point finite-difference scheme. As mentioned earlier, then, the derived scheme can be made to be fourth-order accurate by taking $\alpha = \frac{4}{3}$.

5.3.3. Node-Centered Compact Diffusion Schemes

Reconstruction methods typically result in an extended stencil with not just neighbors but neighbors in the next level. Compact stencil may be desired for an exact linearization in implicit formulations and/or ease with parallelization. To devise a compact diffusion scheme involving only the nearest neighbors, we must evaluate the gradient within a compact stencil. The following is a good example for illustrating a flexibility of the proposed principle in designing a diffusion scheme.

We now redesign the previous edge-based scheme into a compact scheme by replacing the nodal gradients by a common gradient reconstructed for each edge using immediate neighbors only. Note that it is thus only the definition of the left and right states that will be modified. The interface flux to be evaluated is the same as before, i.e., the flux (5.25). In Figure 5.1, for the edge $\{j, k\}$, we reconstruct a linear solution centered at the edge-midpoint by a least-squares method involving five nodes: j , k , k_ℓ , k_r , and k_{rr} . The least-squares problem is to find the solution value, \tilde{u}_{jk} , and the gradient (two components), $\nabla\tilde{u}_{jk}$, at the edge-midpoint: 5 equations for 3 unknowns, which can be solved straightforwardly. For our purpose, the solution value is not needed; only the gradient is used to define the left and right states in Equation (5.28). Alternatively, we may employ the Green-Gauss gradient computed over the two elements that share the edge $\{j, k\}$. Using the reconstructed gradient at both end nodes of the edge, we thus define the left and right states as

$$\begin{aligned} u_L &= u_j + \frac{1}{2} \nabla\tilde{u}_{jk} \cdot \Delta\mathbf{l}_{jk}, & u_R &= u_k - \frac{1}{2} \nabla\tilde{u}_{jk} \cdot \Delta\mathbf{l}_{jk}, \\ (p, q)_L &= \nabla\tilde{u}_{jk}, & (p, q)_R &= \nabla\tilde{u}_{jk}. \end{aligned} \quad (5.34)$$

Using these values to evaluate the diffusive flux (5.25), we obtain a compact finite-volume diffusion scheme. Note that we have $u_L \neq u_R$ in general, unless the reconstructed linear solution happens to pass through the solutions at j and k . Therefore, the damping term is nonzero and plays its role. We emphasize that although the gradient is reconstructed at the edge-midpoint, it is not used here to compute the physical diffusive flux directly as commonly done. The gradient is used here to define the left and right states, i.e., to extrapolate the solution value from the left and right nodes to the edge-midpoint as above, and then compute the numerical flux (5.25).

For triangular grids, if we employ the Green-Gauss gradient over the two elements sharing the edge, the damping term will vanish identically since the Green-Gauss gradient projected along the edge reduces to the edge-derivative. The resulting scheme corresponds to the one proposed in Ref. [57] by Braaten and Connell for tetrahedral grids (see Appendix D for more details).

It may be instructive, although not necessary for implementation, to derive the diffusive flux resulted from inserting the left and right states defined above into Equation (5.25):

$$\phi_{jk} = -\nu \nabla\tilde{u}_{jk} \cdot \hat{\mathbf{n}}_{jk} - \frac{\nu\alpha}{|\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}|} \left[\frac{u_k - u_j}{\Delta l_{jk}} - \nabla\tilde{u}_{jk} \cdot \hat{\mathbf{e}}_{jk} \right]. \quad (5.35)$$

The second term is the damping term. Clearly, it will vanish if the reconstructed gradient projected along the edge matches the edge derivative. Note that if one takes the common approach and uses the least-squares gradient to evaluate the diffusive flux directly as $-\nu \nabla\tilde{u}_{jk} \cdot \hat{\mathbf{n}}_{jk}$, then the resulting scheme corresponds to the above scheme with $\alpha = 0$, i.e., without the damping term. It can suffer from oscillations as will be shown later. Braaten and Connell [57] take the common approach as well, but their scheme does not suffer. Their success appears due to the use of the Green-Gauss gradient (instead of the least-squares gradient) which cancels the damping term even if formulated so as to ensure high-frequency damping as above. It must be noted, however, that the Braaten-Connell scheme may suffer from checkerboard errors if naively extended to quadrilateral or mixed grids since the Green-Gauss gradient is, then, not guaranteed to match the edge-derivative. In that case, a damping term must be incorporated somehow. In contrast, following the proposed principle, we derived a general (more flexible) formula (5.35) that works with both the least-squares and Green-Gauss gradients on *arbitrary* grids.

5.3.4. Element-Based Compact Diffusion Schemes

Alternatively, the gradient may be reconstructed within each element, and the resulting scheme will still be compact. To discuss this strategy, we rewrite the edge-based scheme in the element-based form (see Figure 5.3):

$$\frac{du_j}{dt} = -\frac{1}{V_j} \sum_{E \in \{E_j\}} [\phi_{j\ell}^E \mathbf{n}_{j\ell}^E + \phi_{jr}^E \mathbf{n}_{jr}^E], \quad (5.36)$$

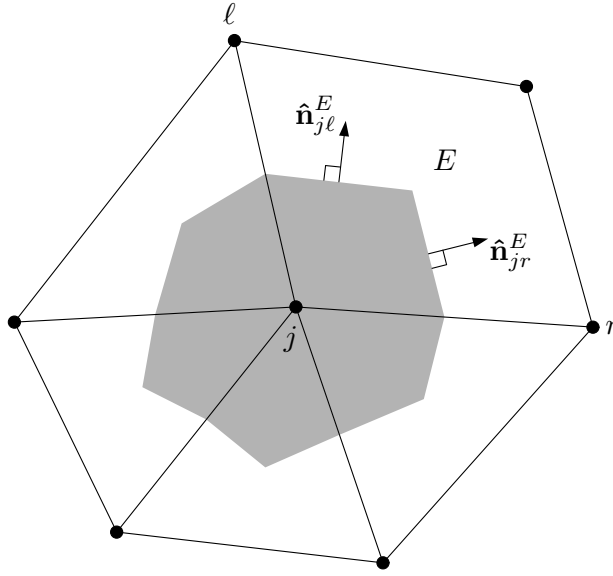


Figure 5.3. Dual control volume for node-centered schemes with unit face-normal vectors associated with an element, E : $\hat{\mathbf{n}}_{j\ell}^E = \mathbf{n}_{j\ell}^E / |\mathbf{n}_{j\ell}^E|$ and $\hat{\mathbf{n}}_{jr}^E = \mathbf{n}_{jr}^E / |\mathbf{n}_{jr}^E|$.

where $\mathbf{n}_{j\ell}^E$ and \mathbf{n}_{jr}^E are scaled normal vectors of the control volume boundary edges, and the superscript, E , has been introduced for the fluxes to indicate the element in which they are evaluated. Here, we still consider evaluating the interface flux at the midpoint of each edge by two extrapolated states (5.28). But we consider reconstructing the gradient not at each node but over each element: a common gradient is employed at two end nodes of each edge. This approach is similar to the interfacial reconstruction in one dimension discussed in Section 4.1.4.

For a triangular element, the solution can be interpolated linearly, giving a unique gradient corresponding to the Green-Gauss gradient, denoted here by $(\nabla u)^E$. By definition, this gradient reduces to the edge-derivative along each edge. This means that the reconstructed solution becomes continuous along each edge: for the edge $\{j, \ell\}$,

$$\begin{aligned} u_L &= u_j + \frac{1}{2} (\nabla u)^E \cdot \Delta \mathbf{l}_{j\ell} = \frac{u_\ell + u_j}{2}, & u_R &= u_\ell - \frac{1}{2} (\nabla u)^E \cdot \Delta \mathbf{l}_{j\ell} = \frac{u_\ell + u_j}{2}, \\ (p, q)_L &= (\nabla u)^E, & (p, q)_R &= (\nabla u)^E, \end{aligned} \quad (5.37)$$

and similarly for the other edge $\{j, r\}$. Therefore the damping term in the diffusion flux (5.25) vanishes identically, and we are left with the consistent part:

$$\phi_{j\ell} = -\nu (\nabla u)^E \cdot \mathbf{n}_{j\ell}^E, \quad (5.38)$$

$$\phi_{jr} = -\nu (\nabla u)^E \cdot \mathbf{n}_{jr}^E. \quad (5.39)$$

The resulting diffusion scheme is equivalent to the Galerkin discretization (also known as the Green-Gauss discretization) just like the one-dimensional scheme in Section 4.1.4.

On the other hand, for a quadrilateral element, the damping term does not necessarily vanish. Again, we employ the Green-Gauss gradient, $(\nabla u)^E$, computed over a quadrilateral. Then we use it to obtain the left and right states:

$$\begin{aligned} u_L &= u_j + \frac{1}{2} (\nabla u)^E \cdot \Delta \mathbf{l}_{j\ell}, & u_R &= u_\ell - \frac{1}{2} (\nabla u)^E \cdot \Delta \mathbf{l}_{j\ell}, \\ (p, q)_L &= (\nabla u)^E, & (p, q)_R &= (\nabla u)^E, \end{aligned} \quad (5.40)$$

for the edge $\{j, \ell\}$, and similarly for the other edge $\{j, r\}$. It must be noted here that the solution along the edge is not necessarily continuous because the projection of the Green-Gauss gradient along the edge does not necessarily match the edge derivative unless the solution values at the four vertices of the quadrilateral happen

to lie in the same plane. The resulting diffusion scheme is therefore a compact scheme with a damping term. To see this, we may insert these interface values into the diffusive flux (5.25) to derive

$$\phi_{j\ell} = -\nu (\nabla u)^E \cdot \mathbf{n}_{E\ell} - \frac{\nu\alpha}{|\hat{\mathbf{e}}_{j\ell} \cdot \hat{\mathbf{n}}_{j\ell}^E|} \left[\frac{u_\ell - u_j}{\Delta l_{j\ell}} - (\nabla u)^E \cdot \hat{\mathbf{e}}_{j\ell} \right], \quad (5.41)$$

$$\phi_{jr} = -\nu (\nabla u)^E \cdot \mathbf{n}_{Er} - \frac{\nu\alpha}{|\hat{\mathbf{e}}_{jr} \cdot \hat{\mathbf{n}}_{jr}^E|} \left[\frac{u_r - u_j}{\Delta l_{jr}} - (\nabla u)^E \cdot \hat{\mathbf{e}}_{jr} \right], \quad (5.42)$$

where $\hat{\mathbf{n}}_{j\ell}^E$ and $\hat{\mathbf{n}}_{jr}^E$ are the unit vectors. The second term is the damping term. Note that the edge-term is, again, incorporated automatically.

We again choose $\alpha = 1$, but for the quadrilateral-element scheme, there is a special choice:

$$\alpha = \begin{cases} (\hat{\mathbf{e}}_{j\ell} \cdot \hat{\mathbf{n}}_{j\ell}^E) |\hat{\mathbf{e}}_{j\ell} \cdot \hat{\mathbf{n}}_{j\ell}^E| & \text{for } \phi_{j\ell}, \\ (\hat{\mathbf{e}}_{jr} \cdot \hat{\mathbf{n}}_{jr}^E) |\hat{\mathbf{e}}_{jr} \cdot \hat{\mathbf{n}}_{jr}^E| & \text{for } \phi_{jr}, \end{cases} \quad (5.43)$$

which reproduces a widely-used element-based finite-volume diffusion scheme (often called the Green-Gauss scheme) [11,54]. This scheme is constructed by deliberately incorporating the edge-term to damp high-frequency modes associated with quadrilateral elements (the checkerboard error modes). Note that similarly to the Avg-LSQ-EN scheme in Section 5.3.2, we see that the damping coefficient of this scheme will be vanishingly small for highly-skewed grids. Alternatively, we may set

$$\alpha = \begin{cases} \frac{|\hat{\mathbf{e}}_{j\ell} \cdot \hat{\mathbf{n}}_{j\ell}^E|}{\hat{\mathbf{e}}_{j\ell} \cdot \hat{\mathbf{n}}_{j\ell}^E} & \text{for } \phi_{j\ell}, \\ \frac{|\hat{\mathbf{e}}_{jr} \cdot \hat{\mathbf{n}}_{jr}^E|}{\hat{\mathbf{e}}_{jr} \cdot \hat{\mathbf{n}}_{jr}^E} & \text{for } \phi_{jr}, \end{cases} \quad (5.44)$$

to derive a scheme analogous to the Avg-LSQ-FT scheme in Section 5.3.2. This scheme is expected to have a large damping coefficient for highly-skewed grids, but it may go negative for excessively-skewed grids beyond 90° . On the other hand, our diffusion scheme has a large positive damping coefficient for any positive real value of α on arbitrarily-skewed grids.

5.4. Residual-Distribution Diffusion Schemes

The residual-distribution method is a good example for illustrating the general applicability of the proposed principle. This example shows that we are not merely proposing a new diffusive flux.

5.4.1. LDA Diffusion Scheme

We consider constructing residual-distribution schemes for diffusion on unstructured triangular grids. Following the general principle, we begin by discretizing the hyperbolic diffusion system. In the residual-distribution method, we first define the cell-residual over a cell T (see Figure 5.4):

$$\Phi^T = \int_T (-\mathbf{A}\mathbf{U}_x - \mathbf{B}\mathbf{U}_y + \mathbf{Q}) dV. \quad (5.45)$$

Assuming a piecewise linear variation of \mathbf{U} over the cell, we obtain

$$\Phi^T = -\sum_{i=1}^3 \mathbf{K}_i \mathbf{U}_i + \bar{\mathbf{Q}}_T V_T, \quad (5.46)$$

where

$$\mathbf{K}_i = \frac{1}{2} (\mathbf{A}, \mathbf{B}) \cdot \mathbf{n}_i, \quad \bar{\mathbf{Q}}_T = \frac{\mathbf{Q}_1 + \mathbf{Q}_2 + \mathbf{Q}_3}{3}, \quad (5.47)$$

$\mathbf{n}_i = (n_{i_x}, n_{i_y})$ is the scaled inward normal (see Figure 5.5), and V_T is the cell area. The derivatives, \mathbf{U}_x and \mathbf{U}_y , have been evaluated by the Green-Gauss integration over the cell which is exact for linear functions. The

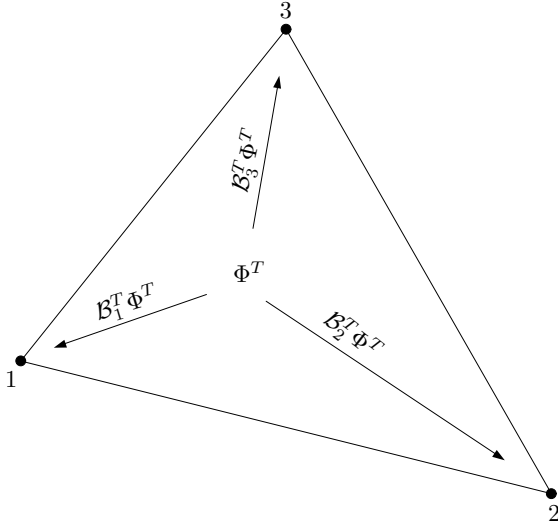


Figure 5.4. Distribution of a cell-residual to the set of vertices $\{i_T\} = \{1, 2, 3\}$. Each contribution is determined by multiplying the cell-residual by the distribution matrix, \mathcal{B}_i^T , where $i \in \{i_T\}$.

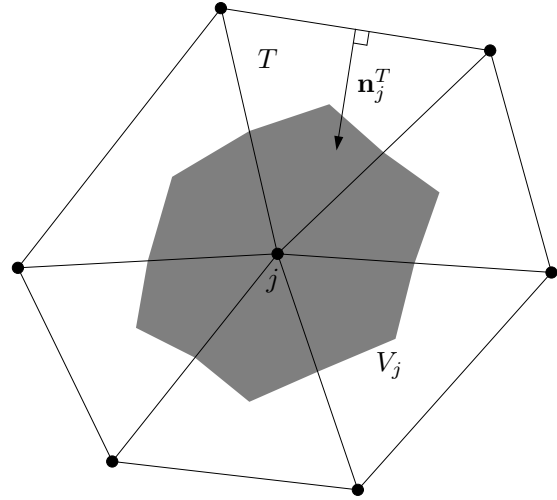


Figure 5.5. Median dual cell around a node j over the set of surrounding triangles, $\{T_j\}$. V_j is the dual cell area. \mathbf{n}_j^T is the scaled inward normal (not drawn to scale) associated with a triangle $T \in \{T_j\}$.

source term has been discretized to be also exact for linear functions, but it is not important since we will ignore it. The cell-residual is now distributed to the nodes of the cell by a distribution matrix, \mathcal{B}_j . In choosing the distribution matrix, we must remember that the system being discretized is hyperbolic. Hence, we employ an upwind distribution matrix. The LDA scheme [43] is a widely-used upwind residual-distribution scheme, which is defined by the following distribution matrix:

$$\mathcal{B}_i^T = \mathbf{K}_i^+ \left(\sum_{i=1}^3 \mathbf{K}_i^+ \right)^{-1}, \quad (5.48)$$

where

$$\mathbf{K}_i^+ = \frac{1}{2} \mathbf{A}_{n_i}^+ = \frac{1}{2} \mathbf{R}_{n_i} \mathbf{\Lambda}_{n_i}^+ \mathbf{R}_{n_i}^{-1} = \frac{\alpha \nu |\mathbf{n}_i|}{L_r} \frac{1}{4} \begin{bmatrix} 1 & -\frac{L_r}{\alpha} \hat{n}_{x_i} & -\frac{L_r}{\alpha} \hat{n}_{y_i} \\ -\frac{\alpha}{L_r} \hat{n}_{x_i} & \hat{n}_{x_i}^2 & \hat{n}_{x_i} \hat{n}_{y_i} \\ -\frac{\alpha}{L_r} \hat{n}_{y_i} & \hat{n}_{x_i} \hat{n}_{y_i} & \hat{n}_{y_i}^2 \end{bmatrix}, \quad (5.49)$$

$\hat{\mathbf{n}}_i = (\hat{n}_{x_i}, \hat{n}_{y_i})$ is the unit inward normal vector, \mathbf{R}_{n_i} is the right-eigenvector matrix of \mathbf{A}_{n_i} , and $\mathbf{\Lambda}_{n_i}^+$ is the corresponding diagonal matrix with positive eigenvalues only. This distribution matrix is often computed numerically, but it can be obtained analytically for the hyperbolic diffusion system:

$$\mathcal{B}_i^T = \begin{bmatrix} \beta_i^T & \frac{L_r}{\alpha} D_{x_i} & -\frac{L_r}{\alpha} D_{y_i} \\ -\frac{\alpha n_{x_i}}{n_T L_r} & -\hat{n}_{x_i} D_{x_i} & \hat{n}_{x_i} D_{y_i} \\ -\frac{\alpha n_{y_i}}{n_T L_r} & -\hat{n}_{y_i} D_{x_i} & \hat{n}_{y_i} D_{y_i} \end{bmatrix}, \quad (5.50)$$

where

$$\beta_i^T = \frac{|\mathbf{n}_i|}{n_T}, \quad n_T = |\mathbf{n}_1| + |\mathbf{n}_2| + |\mathbf{n}_3|, \quad (5.51)$$

$$D_{x_i} = \frac{\hat{n}_{y_k} - \hat{n}_{y_j}}{\hat{n}_{x_1}(\hat{n}_{y_2} - \hat{n}_{y_3}) + \hat{n}_{x_2}(\hat{n}_{y_3} - \hat{n}_{y_1}) + \hat{n}_{x_3}(\hat{n}_{y_1} - \hat{n}_{y_2})}, \quad (5.52)$$

$$D_{y_i} = \frac{\hat{n}_{x_k} - \hat{n}_{x_j}}{\hat{n}_{x_1}(\hat{n}_{y_2} - \hat{n}_{y_3}) + \hat{n}_{x_2}(\hat{n}_{y_3} - \hat{n}_{y_1}) + \hat{n}_{x_3}(\hat{n}_{y_1} - \hat{n}_{y_2})}, \quad (5.53)$$

and $(i, j, k) = (1, 2, 3), (2, 3, 1),$ or $(3, 1, 2)$. After performing the distribution step all over the cells, we have the following semi-discrete equation at each node:

$$\frac{d\mathbf{U}_j}{dt} = \frac{1}{V_j} \sum_{T \in \{T_j\}} \mathcal{B}_j^T \Phi^T, \quad (5.54)$$

where $\{T_j\}$ denotes a set of triangles that share the node j and V_j is the median dual cell area (see Figure 5.5). Note that the distribution matrix sums up to the identity matrix over a cell, so that the sum of the distributed partial cell-residuals reduces to the total cell-residual:

$$\sum_{i=1}^3 \mathcal{B}_i^T \Phi^T = \Phi^T. \quad (5.55)$$

This ensures a discrete conservation: a global sum of the solution changes are solely due to events on the domain boundary. We must point out here that the above semi-discrete formulation may not be time-accurate because the right hand side may be inconsistent by a constant factor with the spatial part of the time-dependent hyperbolic diffusion system (5.3). This is not important for steady computations, but it is crucial for unsteady computations. A consistent semi-discrete form is given by

$$\frac{d\mathbf{U}_j}{dt} = \left(\sum_{T \in \{T_j\}} \mathcal{B}_j^T V_T \right)^{-1} \sum_{T \in \{T_j\}} \mathcal{B}_j^T \Phi^T. \quad (5.56)$$

This is a semi-discrete form at least first-order accurate in time. See Refs. [58, 59, 60] for more sophisticated discussions on time-accurate residual-distribution schemes. Time-accurate computation is now possible by integrating Equation (5.56) by the forward-Euler time-stepping. The time step may be chosen as the minimum of the local time step, Δt_j , defined at each node by

$$\Delta t_j \leq \frac{V_j}{\sum_{T \in \{T_j\}} \frac{1}{2} \sqrt{\frac{\nu}{T_r}} |\mathbf{n}_j^T|}, \quad (5.57)$$

where \mathbf{n}_j^T is the scaled inward normal of the edge opposite to the node j . Having completed the discretization of the hyperbolic diffusion system, we now derive a diffusion scheme by discarding the second and third components:

$$\frac{du_j}{dt} = \frac{1}{\sum_{T \in \{T_j\}} \beta_j^T V_T} \sum_{T \in \{T_j\}} \left[\beta_j^T \phi^T - \frac{\alpha \nu V_T}{L_r} \{ \nabla u^T - (\bar{p}_T, \bar{q}_T) \} \cdot (D_{x_j}, -D_{y_j}) \right], \quad (5.58)$$

where we take $\alpha = 1$ as before, and ϕ^T is the cell-residual for the diffusion equation,

$$\phi^T = \nu [(p_x)^T + (q_y)^T] S^T. \quad (5.59)$$

Here, $(p_x)^T$ and $(q_y)^T$ are the Green-Gauss gradients of p and q over the cell computed with nodal gradients reconstructed, for example, by the Green-Gauss formula (5.27). The gradient reconstruction is necessary because we have lost the update equations for the gradient variables, and the gradient variables are not stored any more at nodes as unknowns. Note that we have retained only the first diagonal entry of $\left(\sum_{T \in \{T_j\}} \mathcal{B}_j^T V_T \right)^{-1}$. The scheme is still time-consistent because only the term proportional to the cell-residual, ϕ^T , is responsible for

the consistency. The second term vanishes when summed over the cell, and thus does not contribute the total cell-residual. This implies that it does not affect the conservation and the consistency of the scheme. In fact, the second term acts as damping. Without this term (e.g., if $\alpha = 0$), the scheme will fail to damp out high-frequency errors. As will be demonstrated by numerical experiments, a lack of damping leads to large solution errors. The time step for the derived diffusion scheme may be chosen as in Equation (5.57). A more precise definition would be

$$\Delta t_j \leq \frac{\sum_{T \in \{T_j\}} \beta_j^T V_T}{\sum_{T \in \{T_j\}} \frac{1}{2} \sqrt{\frac{\nu}{T_r}} |\mathbf{n}_j^T|}. \quad (5.60)$$

The relaxation time, T_r , is defined as in Equation (5.10). The length scale L_r may be obtained based on a stronger cell-wise version of this CFL condition: equating the maximum time step and the relaxation time,

$$\frac{2\beta_j^T V_T}{\sqrt{\frac{\nu}{T_r}} |\mathbf{n}_j^T|} = \alpha T_r, \quad (5.61)$$

solving it for T_r and comparing the result with Equation (5.10), we obtain

$$L_r = \frac{2V_T}{|\mathbf{n}_1| + |\mathbf{n}_2| + |\mathbf{n}_3|}. \quad (5.62)$$

This is precisely the radius of the inscribed circle in the triangular cell. Alternatively, we may take simply a half of the minimum height of the triangle:

$$L_r = \frac{1}{2} h_{min} = \frac{V_T}{\max(|\mathbf{n}_1|, |\mathbf{n}_2|, |\mathbf{n}_3|)}. \quad (5.63)$$

For numerical experiments, we use this simpler version, but results are very similar to those obtained with Equation (5.62).

5.4.2. Lax-Wendroff Diffusion Scheme

Instead of the LDA scheme, we may employ the Lax-Wendroff scheme defined by the distribution matrix [32, 43, 61, 62],

$$\mathcal{B}_i^T = \frac{1}{3} \mathbf{I} + \frac{\tau}{2V_T} \mathbf{K}_i, \quad (5.64)$$

where \mathbf{I} is the identity matrix and τ is a time-like parameter. The parameter τ may be defined as the maximum time step expressed in terms of the relaxation time (see Equation (3.6)):

$$\tau = \alpha T_r. \quad (5.65)$$

Applying the Lax-Wendroff matrix to the semi-discrete equation (5.56), and then ignoring the second component, we obtain the following diffusion scheme:

$$\frac{du_j}{dt} = \frac{1}{V_j} \sum_{T \in \{T_j\}} \left[\frac{1}{3} \phi^T - \frac{\nu \alpha}{4} \{ \nabla u^T - (\bar{p}_T, \bar{q}_T) \} \cdot \mathbf{n}_j^T \right]. \quad (5.66)$$

Again, it is the second term that acts as damping. The failure of the isotropic scheme, which corresponds to taking $\alpha = 0$, has already been pointed out in Ref. [21]: insufficient high-frequency damping and extremely slow convergence to a steady state. We remark that as proved in Ref. [32], this scheme reduces to the Galerkin scheme when $\alpha = 2$. Even in that case, this scheme may still be useful for integrating the Galerkin scheme with an advection scheme to construct a uniformly accurate residual-distribution scheme for the advection-diffusion equation, in a manner suggested in Refs. [9, 34]. To see if there is a special choice for α , expand the Lax-Wendroff diffusion scheme (5.66) on a Cartesian grid of spacing h with regularly inserted diagonals to get

$$\frac{du_j}{dt} = \nu(u_{xx} + u_{yy}) - \frac{\nu}{24} [(3\alpha - 8)(u_{xxxx} + u_{yyyy}) + 2(\alpha - 2)(u_{xxxy} + 2u_{xyxy} + u_{xxyy})] h^2 + O(h^4). \quad (5.67)$$

It follows that no single value of α can make the leading error term vanish, and therefore the scheme is second-order accurate at best. But as the choice $\alpha = 2$ eliminates the cross-derivative error terms, the choice $\alpha = 8/3$ eliminates the other group of error terms (those of the Galerkin scheme). Numerical experiments show that the Lax-Wendroff-type scheme can produce only slightly more accurate solutions than the Galerkin scheme with this choice. A similar study for the LDA diffusion scheme shows that there exist no positive values of α that eliminate any part of the leading second-order error term.

5.5. Cell-Centered Finite-Volume Diffusion Schemes

We consider constructing cell-centered diffusion schemes. Without loss of generality, we consider only triangular grids. Resulting diffusion schemes can be easily applied to other types of grids.

5.5.1. Face-Midpoint Scheme

A domain of interest is divided into a set of non-overlapping triangular cells. The solution values are stored at centroids of triangular cells and considered as cell-averaged values. Evolution of a cell-averaged value, \mathbf{U}_j , is determined by approximating the integral of the hyperbolic diffusion system over the corresponding triangular cell, T_j :

$$\frac{d\mathbf{U}_j}{dt} = \frac{1}{V_j} \int_{T_j} (-\mathbf{F}_x - \mathbf{G}_y + \mathbf{Q}) dV \quad (5.68)$$

$$= -\frac{1}{V_j} \oint_{\partial T_j} \mathbf{H} dA + \int_{T_j} \mathbf{Q} dV, \quad (5.69)$$

where V_j here denotes the volume of the cell T_j , ∂T_j denotes the cell boundary, dA is the infinitesimal area of the boundary, and \mathbf{H} is the flux dotted with the unit outward normal vector of the boundary, $\hat{\mathbf{n}}$. We evaluate the flux integral by the midpoint rule over each face of the triangle; a face is associated with a pair of adjacent data points, j and k (see Figure 5.6). Replacing \mathbf{H} by a numerical flux, we thus obtain the following semi-discrete form:

$$\frac{d\mathbf{U}_j}{dt} = -\frac{1}{V_j} \sum_{k \in \{K_j\}} \Phi_{jk} A_{jk} + \frac{1}{V_j} \int_{T_j} \mathbf{Q} dV, \quad (5.70)$$

where $\{K_j\}$ is a set of faces surrounding the data point j , and A_{jk} is the face area. The numerical flux Φ_{jk} is the upwind flux which is essentially the same as the one employed in the node-centered scheme:

$$\Phi_{jk} = \frac{1}{2} [\mathbf{H}_{jk}(\mathbf{U}_R) + \mathbf{H}_{jk}(\mathbf{U}_L)] - \frac{1}{2} |\mathbf{A}_n| (\mathbf{U}_R - \mathbf{U}_L), \quad (5.71)$$

where \mathbf{H}_{jk} is the physical flux projected along the outward face-normal direction,

$$\mathbf{H}_{jk} = [\mathbf{F}, \mathbf{G}] \cdot \hat{\mathbf{n}}_{jk}, \quad (5.72)$$

Note that $\hat{\mathbf{n}}_{jk}$ is the unit face-normal vector. For second-order accuracy, the solution gradients are reconstructed in each cell, and the left and right states, \mathbf{U}_L and \mathbf{U}_R , are extrapolated from each cell to the face midpoint (see Figure 5.6). Time integration may be performed by the forward-Euler time-stepping scheme with a time step restricted by a CFL condition similar to the node-centered finite-volume scheme discussed in Section 5.3.2.

Having completed the discretization of the hyperbolic diffusion system, we now discard the second and third components and obtain a diffusion scheme:

$$\frac{du_j}{dt} = -\frac{1}{V_j} \sum_{k \in \{K_j\}} \phi_{jk} A_{jk}, \quad (5.73)$$

where

$$\phi_{jk} = -\frac{\nu}{2} [(p, q)_R + (p, q)_L] \cdot \hat{\mathbf{n}}_{jk} - \frac{1}{2} \sqrt{\frac{\nu}{T_r}} (u_R - u_L). \quad (5.74)$$

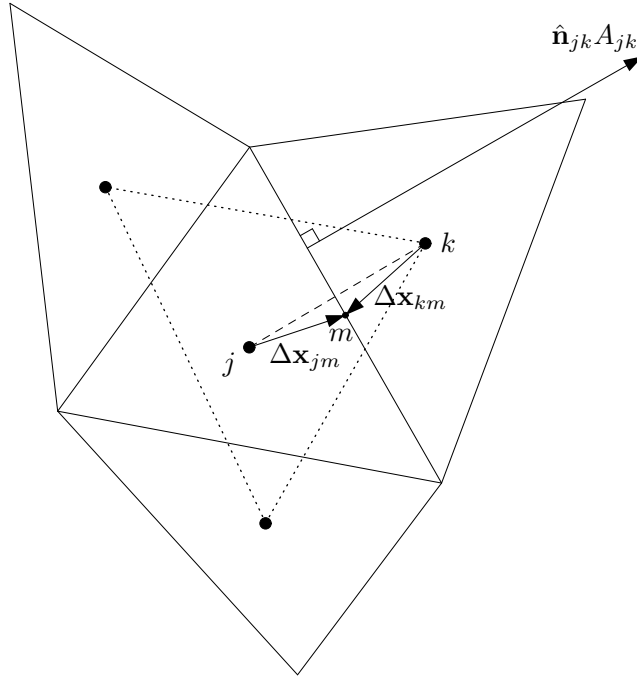


Figure 5.6. Stencil for cell-centered schemes.

The dissipation term of the upwind flux has now entered into the diffusive flux as a damping term. It has been automatically introduced with no specific technique to incorporate it, and its coefficient is determined unambiguously. The relaxation time, T_r , is defined as in Equation (5.10), thus yielding

$$\phi_{jk} = -\frac{\nu}{2} [(p, q)_R + (p, q)_L] \cdot \hat{\mathbf{n}}_{jk} - \frac{\nu\alpha}{2L_r} (u_R - u_L). \quad (5.75)$$

The length scale, L_r , is defined based on the distance between the two data points across the face projected along the face-normal direction:

$$L_r = \frac{1}{2} |\Delta \mathbf{l}_{jk} \cdot \hat{\mathbf{n}}_{jk}|, \quad (5.76)$$

where $\Delta \mathbf{l}_{jk} = (x_k - x_j, y_k - y_j)$. As in the edge-based schemes in Section 5.3.2, the above definition of L_r brings the skewness measure in the denominator of the damping coefficient. For the parameter α , we again set $\alpha = 1$, but also consider using $\alpha = 4/3$ which makes the scheme fourth-order accurate on structured grids. To define the left and right states, we evaluate the gradient variables by reconstructed solution gradients in cells, ∇u_k and ∇u_j . Then, linearly extrapolating the solution to the face-midpoint, we obtain

$$\begin{aligned} u_L &= u_j + \nabla u_j \cdot \Delta \mathbf{x}_{jm}, & u_R &= u_k + \nabla u_k \cdot \Delta \mathbf{x}_{km}, \\ (p, q)_L &= \nabla u_j, & (p, q)_R &= \nabla u_k. \end{aligned} \quad (5.77)$$

The diffusion scheme has thus been completely defined. Note that there is no need to store p and q as they are explicitly evaluated as above. For the gradient reconstruction in each cell, we define a triangle formed by neighbor centroids as shown by dotted lines in Figure 5.6, and determine a unique gradient by interpolation and differentiation or equivalently by the Green-Gauss integration over that triangle. If a triangle has only two valid neighbors (i.e., one boundary face), we add the triangle itself. If it has only one valid neighbor, we simply copy the reconstructed gradient of that neighbor. For arbitrary grids, a cell may have more than three neighbors (e.g., a quadrilateral has four neighbors); then a least-squares gradient reconstruction method may be employed. For highly-stretched grids, gradients reconstructed using only the neighbors are not generally accurate [63]. In such a case, we use a least-squares method using all cells sharing nodes of the cell in which the gradient is sought. It should be noted again that the derived cell-centered diffusion scheme can be implemented in the same framework of advection schemes: reconstruction, extrapolation, and interface flux evaluation.

5.5.2. Edge-Midpoint Scheme

It is also possible to evaluate the interface flux at the edge-midpoint, i.e., halfway between the two adjacent data points, rather than the face-midpoint:

$$\begin{aligned} u_L &= u_j + \frac{1}{2} \nabla u_j \cdot \Delta \mathbf{l}_{jk}, & u_R &= u_k - \frac{1}{2} \nabla u_k \cdot \Delta \mathbf{l}_{jk}, \\ (p, q)_L &= \nabla u_j, & (p, q)_R &= \nabla u_k. \end{aligned} \quad (5.78)$$

Since the edge-midpoint is not necessarily located on the face (i.e., off the control volume boundary), the diffusive flux may not be evaluated precisely on the control volume boundary. However, it does not affect the formal order of accuracy of the scheme, at least, for second-order accurate schemes considered here. This is because the gradient is constant over each cell, so that the consistent part, which is relevant to the order of accuracy, does not depend on the location where it is evaluated. It only brings a slight change in the amount of damping.

We remark that the above scheme is formally identical to the edge-based node-centered finite-volume scheme (5.29), and widely-used cell-centered diffusion schemes can be reproduced in the same way as before: the Avg-LSQ-EN scheme with Equation (5.30), and the Avg-LSQ-FT scheme with Equation (5.33). Again, the damping coefficient can be vanishingly small for the edge-normal case and can be large for the face-tangent case. It should be pointed out that in both cases, the damping coefficient may even go negative especially on agglomerated (arbitrary polygonal) grids typically used in multigrid methods [56,64,65,66]. In contrast, the damping coefficient of our diffusion scheme is guaranteed to be positive for any positive real value of α .

5.6. Discontinuous-Galerkin Diffusion Schemes

We consider constructing P_1 DG diffusion schemes on triangular grids. Following the principle, we begin by discretizing the first-order hyperbolic diffusion system. Within a cell, T_j , we express the solution as a polynomial of degree 1 centered at the centroid $\mathbf{x}_c = (x_j, y_j)$:

$$\mathbf{U}(x, y) = \psi_0 \mathbf{U}_j + \psi_1 \overline{\partial_x \mathbf{U}_j} + \psi_2 \overline{\partial_y \mathbf{U}_j}, \quad (5.79)$$

where \mathbf{U}_j is a vector of the cell-averaged solutions, and $\overline{\partial_x \mathbf{U}_j}$ and $\overline{\partial_y \mathbf{U}_j}$ are vectors of the x - and y -derivatives, respectively:

$$\mathbf{U}_j = \begin{bmatrix} u_j \\ p_j \\ q_j \end{bmatrix}, \quad \overline{\partial_x \mathbf{U}_j} = \begin{bmatrix} \overline{\partial_x u_j} \\ \overline{\partial_x p_j} \\ \overline{\partial_x q_j} \end{bmatrix}, \quad \overline{\partial_y \mathbf{U}_j} = \begin{bmatrix} \overline{\partial_y u_j} \\ \overline{\partial_y p_j} \\ \overline{\partial_y q_j} \end{bmatrix}. \quad (5.80)$$

It should be noted that the degrees of freedom for p and q have been introduced here just for the sake of convenience; they will be abandoned once the discretization of the hyperbolic diffusion system is completed. The functions, ψ_m , $m = 0, 1, 2$, are the basis functions,

$$\psi_0 = 1, \quad \psi_1 = x - x_j, \quad \psi_2 = y - y_j, \quad (5.81)$$

specifically chosen such that ψ_0 is orthogonal to the other two:

$$\int_{T_j} \psi_0 \psi_m dV = 0, \quad m = 1, 2. \quad (5.82)$$

Evolution equations for the unknowns, \mathbf{U}_j , $\overline{\partial_x \mathbf{U}_j}$, and $\overline{\partial_y \mathbf{U}_j}$, are derived from the weak formulation involving a test function v :

$$\int_{T_j} v \frac{\partial \mathbf{U}}{\partial t} dV = \int_{T_j} v (-\mathbf{F}_x - \mathbf{G}_y + \mathbf{Q}) dV \quad (5.83)$$

$$= - \oint_{\partial T_j} v \mathbf{H} dA + \int_{T_j} \nabla v \cdot (\mathbf{F}, \mathbf{G}) dV + \int_{T_j} v \mathbf{Q} dV \quad (5.84)$$

$$= - \sum_{k \in \{K_j\}} \int_{\partial T_j^k} v \mathbf{H} dA_k + \int_{T_j} \nabla v \cdot (\mathbf{F}, \mathbf{G}) dV + \int_{T_j} v \mathbf{Q} dV, \quad (5.85)$$

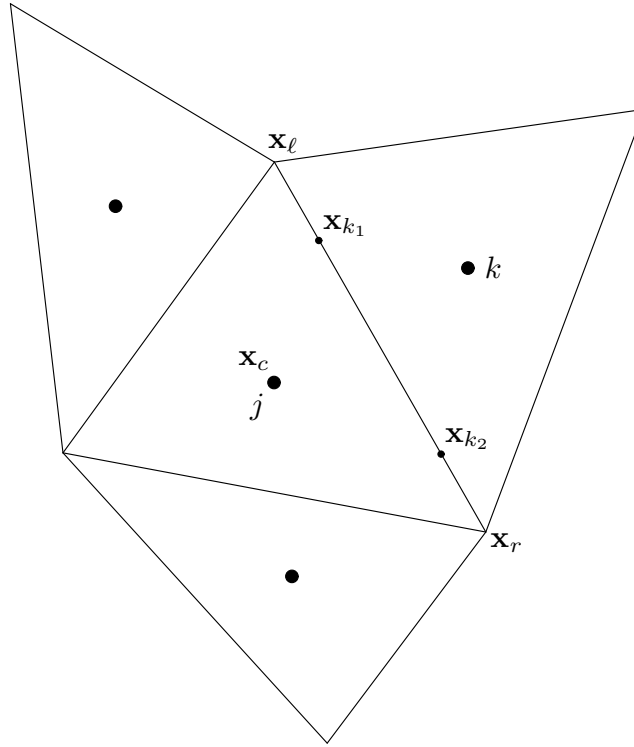


Figure 5.7. Stencil for discontinuous-Galerkin schemes. Quadrature points are shown for the k -th face.

where ∂T_j^k denotes the boundary corresponding to the k -th face, and \mathbf{H} is the flux normal to the boundary. Setting $v = \psi_0$, we obtain a decoupled evolution equation for the cell-average:

$$V_j \frac{d\mathbf{U}_j}{dt} = - \sum_{k \in \{K_j\}} \int_{\partial T_j^k} \mathbf{H} dA_k + \int_{T_j} \mathbf{Q} dV. \quad (5.86)$$

On the other hand, by setting $v = \psi_1$ and $v = \psi_2$, we obtain a coupled system for the gradients:

$$\begin{bmatrix} \frac{d(\overline{\partial_x \mathbf{U}_j})}{dt} \\ \frac{d(\overline{\partial_y \mathbf{U}_j})}{dt} \end{bmatrix} = M_j^{-1} \begin{bmatrix} - \sum_{k \in \{K_j\}} \int_{\partial T_j^k} \psi_1 \mathbf{H} dA_k + \int_{T_j} \nabla \psi_1 \cdot (\mathbf{F}, \mathbf{G}) dV + \int_{T_j} \psi_1 \mathbf{Q} dV \\ - \sum_{k \in \{K_j\}} \int_{\partial T_j^k} \psi_2 \mathbf{H} dA_k + \int_{T_j} \nabla \psi_2 \cdot (\mathbf{F}, \mathbf{G}) dV + \int_{T_j} \psi_2 \mathbf{Q} dV \end{bmatrix}. \quad (5.87)$$

The mass matrix, M_j , which has been formally inverted above to decouple the time derivatives, is given by

$$M_j = \begin{bmatrix} \int_{T_j} \psi_1^2 dV & \int_{T_j} \psi_1 \psi_2 dV \\ \int_{T_j} \psi_2 \psi_1 dV & \int_{T_j} \psi_2^2 dV \end{bmatrix} = \frac{V_j}{12} \sum_{i=1}^3 \begin{bmatrix} (x_i - x_j)^2 & (x_i - x_j)(y_i - y_j) \\ (x_i - x_j)(y_i - y_j) & (y_i - y_j)^2 \end{bmatrix}, \quad (5.88)$$

where (x_i, y_i) , $i = 1, 2, 3$, are the vertex coordinates of the triangle. This matrix depends only on the geometry of the grid. Therefore, it can be precomputed and inverted just once, and reused during computations. For the integral that appears on the right hand side of each evolution equation, it is required that the boundary and volume integrals be exact for polynomials of degree three and of degree two (i.e., fourth-order and third-order accurate), respectively [67]. Following Ref. [67], we evaluate the boundary integral with two quadrature points per face (see Figure 5.7):

$$\int_{\partial T_j^k} \psi_s(\mathbf{x}) \mathbf{H} dA_k = \sum_{n=1}^2 w_{k_n} \psi_s(\mathbf{x}_{k_n}) \Phi_{k_n}(\mathbf{x}_{k_n}) A_k, \quad k \in \{K_j\}, \quad s = 0, 1, 2, \quad (5.89)$$

where the normal flux has been replaced by the upwind numerical flux in the face normal direction:

$$\Phi_{k_n}(\mathbf{x}_{k_n}) = \frac{1}{2} [\mathbf{H}(\mathbf{U}_{k_n}^+) + \mathbf{H}(\mathbf{U}_{k_n}^-)] - \frac{1}{2} |\mathbf{A}_n| (\mathbf{U}_{k_n}^+ - \mathbf{U}_{k_n}^-). \quad (5.90)$$

at each quadrature point, where $\mathbf{U}_{k_n}^-$ and $\mathbf{U}_{k_n}^+$ denote the interface states in the internal element and the neighboring element, respectively. The quadrature parameters are given by

$$w_{k_1} = w_{k_2} = \frac{1}{2}, \quad (5.91)$$

$$\mathbf{x}_{k_1} = \frac{1}{2} \left(1 + \frac{1}{\sqrt{3}}\right) \mathbf{x}_\ell + \frac{1}{2} \left(1 - \frac{1}{\sqrt{3}}\right) \mathbf{x}_r, \quad \mathbf{x}_{k_2} = \frac{1}{2} \left(1 - \frac{1}{\sqrt{3}}\right) \mathbf{x}_\ell + \frac{1}{2} \left(1 + \frac{1}{\sqrt{3}}\right) \mathbf{x}_r. \quad (5.92)$$

Now, before we discuss the volume integral, we discard the second and third components from the evolution equations, and obtain the following diffusion scheme:

$$V_j \frac{du_j}{dt} = - \sum_{k \in \{K_j\}} \sum_{n=1}^2 w_{k_n} \phi_{k_n}(\mathbf{x}_{k_n}) A_k, \quad (5.93)$$

$$\begin{bmatrix} \frac{d(\overline{\partial_x u_j})}{dt} \\ \frac{d(\overline{\partial_y u_j})}{dt} \end{bmatrix} = M_j^{-1} \begin{bmatrix} - \sum_{k \in \{K_j\}} \sum_{n=1}^2 w_{k_n} \psi_1(\mathbf{x}_{k_n}) \phi_{k_n}(\mathbf{x}_{k_n}) A_k + \int_{T_j} \nabla \psi_1 \cdot (f, g) dV, \\ - \sum_{k \in \{K_j\}} \sum_{n=1}^2 w_{k_n} \psi_2(\mathbf{x}_{k_n}) \phi_{k_n}(\mathbf{x}_{k_n}) A_k + \int_{T_j} \nabla \psi_2 \cdot (f, g) dV \end{bmatrix}, \quad (5.94)$$

where $\phi_{k_n}(\mathbf{x}_{k_n})$ is the first component of the upwind flux, $\Phi_{k_n}(\mathbf{x}_{k_n})$,

$$\phi_{k_n}(\mathbf{x}_{k_n}) = -\frac{\nu}{2} [(p, q)_{k_n}^+ + (p, q)_{k_n}^-] \cdot \hat{\mathbf{n}}_{jk} - \frac{1}{2} \sqrt{\frac{\nu}{T_r}} (u_{k_n}^+ - u_{k_n}^-). \quad (5.95)$$

The vector $\hat{\mathbf{n}}_{jk}$ is the unit outward normal of the face k . Note that the dissipation term of the upwind flux has entered into the diffusive flux as a damping (or penalty) term, and that its coefficient is determined unambiguously. The relaxation time is defined as in Equation (5.10) based on the distance between the centroids of the two adjacent cells (via a quadrature point) projected along the face-normal vector as in Equation (5.76):

$$L_r = \frac{1}{2} |\Delta \mathbf{l}_{ik} \cdot \hat{\mathbf{n}}_{ik}|, \quad (5.96)$$

where $\Delta \mathbf{l}_{jk} = (x_k - x_j, y_k - y_j)$. For the parameter α , we take $\alpha = 1$ or $\alpha = 3$; the latter corresponds to the fourth-order scheme in one dimension. To evaluate the diffusive flux, we need to define the interface states. In doing so, we need to evaluate the gradient variables, p and q , whose degrees of freedom have been discarded. In the DG method, these gradients may be evaluated simply by differentiating the approximation of the solution. For the P_1 element considered here, it leads to the constant gradients, $\overline{\nabla} u_j = (\overline{\partial_x u_j}, \overline{\partial_y u_j})$ in T_j , and $\overline{\nabla} u_k = (\overline{\partial_x u_k}, \overline{\partial_y u_k})$ in T_k . The interface states at the quadrature point, \mathbf{x}_{k_n} , are then given by

$$\begin{aligned} u_{k_n}^- &= u_j + \overline{\nabla} u_j \cdot (\mathbf{x}_{k_n} - \mathbf{x}_j), & u_{k_n}^+ &= u_k + \overline{\nabla} u_k \cdot (\mathbf{x}_{k_n} - \mathbf{x}_k), \\ (p, q)_{k_n}^- &= \overline{\nabla} u_j, & (p, q)_{k_n}^+ &= \overline{\nabla} u_k. \end{aligned} \quad (5.97)$$

The diffusion scheme thus constructed is compact, involving only the immediate neighbors. No gradient reconstruction is necessary since we carry the solution gradient in each cell. Also, unlike other DG schemes utilizing auxiliary variables, such as the Bassi-Rebay scheme [33] and the LDG scheme [4], the derived diffusion scheme requires no extra variables and careful discretizations of their equations [68]. Numerical experiments show that the derived diffusion scheme are, despite being remarkably simpler, comparably or more accurate than these well-known DG schemes (see Section 6.2). As in one dimension, the derived diffusion scheme is similar to the interior-penalty schemes [22, 45, 46, 47, 48]; but it has distinguished features that it involves no mesh-dependent parameters and even more importantly it involves the skewness measure in the denominator of the damping

coefficient through L_r . The latter has an effect of increasing the damping on highly-skewed grids, enabling robust computations on such grids. Again, it should be noted that the derived diffusion scheme has the same implementation structure as a corresponding advection scheme: interface flux evaluated by two discontinuous states meeting at a quadrature point.

It is interesting to compare the derived diffusion scheme with an interior-penalty scheme proposed by Shahbazi [47] which has an explicit expression for the penalty parameter (no mesh-dependent parameters). We consider its simplified version given in Ref. [69]:

$$\phi_{k_n}^{Shahbazi}(\mathbf{x}_{k_n}) = -\frac{\nu}{2} [(p, q)_{k_n}^+ + (p, q)_{k_n}^-] \cdot \hat{\mathbf{n}}_{jk} - \frac{3\nu}{2} \max\left(\frac{\ell_{T_j}}{V_{T_j}}, \frac{\ell_{T_k}}{V_{T_k}}\right) (u_{k_n}^+ - u_{k_n}^-), \quad (5.98)$$

where ℓ_{T_j} and V_{T_j} are the perimeter and the volume (area) of the element T_j , and similarly ℓ_{T_k} and V_{T_k} are those of the element T_k . On the other hand, we substitute Equation (5.10) into the flux (5.95) to write the derived diffusion scheme as

$$\phi_{k_n}(\mathbf{x}_{k_n}) = -\frac{\nu}{2} [(p, q)_{k_n}^+ + (p, q)_{k_n}^-] \cdot \hat{\mathbf{n}}_{jk} - \frac{\alpha\nu}{2L_r} (u_{k_n}^+ - u_{k_n}^-). \quad (5.99)$$

Comparing the damping coefficients of the two fluxes, we find that the interior-penalty scheme of Shahbazi corresponds to taking

$$\alpha = 3L_r \max\left(\frac{\ell_{T_j}}{V_{T_j}}, \frac{\ell_{T_k}}{V_{T_k}}\right), \quad (5.100)$$

in our scheme; it thus depends on the geometry of two neighboring cells. To estimate this α , we express each quantity in terms of the radius of the inscribed circle in each element, r_{T_j} in T_j and r_{T_k} in T_k :

$$L_r = \frac{1}{6} \left(\frac{\ell_{T_j}}{A_{jk}} r_{T_j} + \frac{\ell_{T_k}}{A_{jk}} r_{T_k} \right), \quad \frac{\ell_{T_j}}{V_{T_j}} = \frac{2}{r_{T_j}}, \quad \frac{\ell_{T_k}}{V_{T_k}} = \frac{2}{r_{T_k}}, \quad (5.101)$$

and we write Equation (5.100) as

$$\alpha = \frac{\frac{\ell_{T_j}}{A_{jk}} r_{T_j} + \frac{\ell_{T_k}}{A_{jk}} r_{T_k}}{\min(r_{T_j}, r_{T_k})}. \quad (5.102)$$

Since $\frac{\ell_{T_j}}{A_{jk}} > 2$ and $\frac{\ell_{T_k}}{A_{jk}} > 2$ for any triangle, we find

$$\alpha = \frac{\frac{\ell_{T_j}}{A_{jk}} r_{T_j} + \frac{\ell_{T_k}}{A_{jk}} r_{T_k}}{\min(r_{T_j}, r_{T_k})} > 4. \quad (5.103)$$

For equilateral triangles, $\frac{\ell_{T_j}}{A_{jk}} = \frac{\ell_{T_k}}{A_{jk}} = 3$ and $r_{T_j} = r_{T_k}$, and therefore we find

$$\alpha = 6. \quad (5.104)$$

For anisotropic triangles, it can be close to 4 at one face, but at the same time can become unlimitedly large (as large as several hundreds) at other faces. Therefore, in general, the interior-penalty scheme of Shahbazi has a very large damping effect compared with ours, $\alpha = 1$ or $\alpha = 3$.

To completely define the diffusion scheme, the volume integrals in Equation (5.87) remain to be discretized. Here we consider two different strategies for evaluating the volume integrals with the required third-order accuracy. Resulting DG schemes will be referred to as DG-I and DG-II schemes.

DG-I Scheme: The first strategy is to use the midpoint of the face, $m_k \in \{K_j\}$, which yields a quadrature of formally third-order accuracy:

$$\int_{T_j} \nabla\psi_1 \cdot (f, g) dV = \int_{T_j} f dV = \sum_{k \in \{K_j\}} \frac{1}{3} f(\mathbf{x}_{m_k}) V_j, \quad (5.105)$$

$$\int_{T_j} \nabla\psi_2 \cdot (f, g) dV = \int_{T_j} g dV = \sum_{k \in \{K_j\}} \frac{1}{3} g(\mathbf{x}_{m_k}) V_j, \quad (5.106)$$

where \mathbf{x}_{m_k} denotes the coordinates of the midpoint m_k . The accuracy of the quadrature, however, deteriorates if the fluxes are evaluated by the solution derivatives in the cell: $f = -\nu p = -\nu \overline{\partial_x u_j}$ and $g = -\nu q = -\nu \overline{\partial_y u_j}$. This is because the gradients are constant over the cell; the quadrature formula reduces to the one-point quadrature formula at the centroid which is only second-order accurate. One possible remedy is to evaluate the flux by the average flux across the cell face:

$$f(\mathbf{x}_{m_k}) = \frac{1}{2} \{f(\mathbf{x}_{m_k})|_j + f(\mathbf{x}_{m_k})|_k\} = -\frac{\nu}{2} \{\overline{\partial_x u_j} + \overline{\partial_x u_k}\}, \quad (5.107)$$

$$g(\mathbf{x}_{m_k}) = \frac{1}{2} \{g(\mathbf{x}_{m_k})|_j + g(\mathbf{x}_{m_k})|_k\} = -\frac{\nu}{2} \{\overline{\partial_y u_j} + \overline{\partial_y u_k}\}, \quad (5.108)$$

so that the flux values are distinct at quadrature points and thus the formal third-order accuracy is recovered. In one dimension, this strategy can be employed straightforwardly (without loss of quadrature accuracy) since the flux values computed for the boundary integral can be directly re-used for the volume integral as we actually did in Section 4.3.

DG-II Scheme: The second strategy is to evaluate the volume integral by integration by parts. It is an advantage of this strategy that the volume integral can be computed within a single cell without using any neighbor information. The physical flux being the gradient of the solution, i.e., $[f, g] = -\nu \nabla u$, we can perform integration by parts to get

$$\int_{T_j} \nabla \psi_1 \cdot [f, g] dV = -\nu \oint_{\partial T_j} u \nabla \psi_1 \cdot \hat{\mathbf{n}}_k dA_k - \int_{T_j} u \nabla^2 \psi_1 dV, \quad (5.109)$$

$$\int_{T_j} \nabla \psi_2 \cdot [f, g] dV = -\nu \oint_{\partial T_j} u \nabla \psi_2 \cdot \hat{\mathbf{n}}_k dA_k - \int_{T_j} u \nabla^2 \psi_2 dV. \quad (5.110)$$

For the P_1 DG method, the second terms vanish, and we are left with the boundary integrals only. Since they are now boundary integrals, we can use the same two-point quadrature formula as before with the solution values (the internal state) already prepared for evaluating the interface flux:

$$\int_{T_j} \nabla \psi_1 \cdot [f, g] dV = -\nu \sum_{k \in \{K_j\}} \sum_{n=1}^2 w_{k_n} u(\mathbf{x}_{k_n}) \nabla \psi_1(\mathbf{x}_{k_n}) \cdot \hat{\mathbf{n}}_k A_k, \quad (5.111)$$

$$\int_{T_j} \nabla \psi_2 \cdot [f, g] dV = -\nu \sum_{k \in \{K_j\}} \sum_{n=1}^2 w_{k_n} u(\mathbf{x}_{k_n}) \nabla \psi_2(\mathbf{x}_{k_n}) \cdot \hat{\mathbf{n}}_k A_k. \quad (5.112)$$

This is genuinely third-order accurate, thus satisfying the accuracy requirement for the volume integral. We remark that here, integration by parts was applied to the volume integral with a solid objective: to evaluate the volume integral with a required accuracy. Interestingly, in one dimension, this strategy leads to the same accuracy degradation problem in quadrature as in the first strategy (DG-I scheme). See Equation (4.64), which corresponds to integration by parts, where averaged solution values at interfaces are actually used, just like we did for the DG-I scheme.

5.7. Spectral-Volume Diffusion Schemes

Here, we consider constructing spectral-volume diffusion scheme on unstructured triangular grids [70]. Similarly to the one-dimensional case, a triangular cell, T , is subdivided systematically into a set of control volumes, $\{C_T\}$, and the solution values are stored within the control volumes as volume-averaged values:

$$\mathbf{U}_i = \frac{1}{V_i} \int_{C_i} \mathbf{U} dV, \quad i \in \{C_T\}, \quad (5.113)$$

where C_i denotes the i -th control volume and V_i is the volume of C_i . The triangular cell thus partitioned is called the spectral volume. Given volume-averaged solutions, we construct a polynomial, \mathbf{U}_T , of degree m :

$$\mathbf{U}_T(x, y) = \sum_{i \in \{C_T\}} \mathbf{U}_i L_i(x, y), \quad (5.114)$$

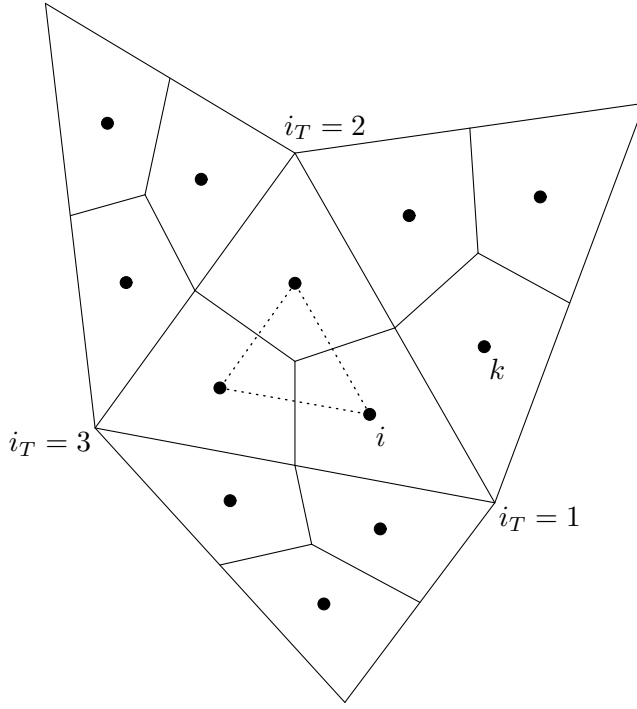


Figure 5.8. Stencil for spectral-volume schemes. Dots indicate the data points defined by the averaged coordinates of the control volumes.

where $L_i(x, y)$ is a shape function that satisfies

$$\frac{1}{V_i} \int_{C_i} \mathbf{U}_T dV = \mathbf{U}_i, \quad i \in \{C_T\}. \quad (5.115)$$

The number of control volumes, N , within a spectral volume is determined by the polynomial degree [70]: $N = (m + 1)(m + 2)/2$. Evolution equation for the volume-averaged value is obtained by integrating the system over the control volume:

$$\frac{d\mathbf{U}_i}{dt} = \frac{1}{V_i} \int_{C_i} (-\mathbf{F}_x - \mathbf{G}_y + \mathbf{Q}) dV \quad (5.116)$$

$$= -\frac{1}{V_i} \oint_{\partial C_i} \mathbf{H} dA + \frac{1}{V_i} \int_{C_i} \mathbf{Q} dV \quad (5.117)$$

$$= -\frac{1}{V_i} \sum_{k \in \{K_i\}} \int_{\partial C_i^k} \mathbf{H} dA + \frac{1}{V_i} \int_{C_i} \mathbf{Q} dV, \quad (5.118)$$

where $\{K_i\}$ is a set of faces of C_i . The boundary integral over each face is discretized by an $(m + 1)^{th}$ order accurate quadrature:

$$\int_{\partial C_i^k} \mathbf{H} dA = \sum_{n=1}^{N_p} w_n \Phi_{ik}(\mathbf{x}_n) A_k, \quad (5.119)$$

where N_p is the number of quadrature points, w_n is the weight at the n -th point, \mathbf{x}_n , and A_k is the area of the k -th face. Note also that the boundary flux, \mathbf{H} , has been replaced by the upwind numerical flux (5.90) at each quadrature point. Now, by ignoring the second and third components, we derive a diffusion scheme:

$$\frac{du_i}{dt} = -\frac{1}{V_i} \sum_{k \in \{K_i\}} \sum_{n=1}^{N_p} w_n \phi_{ik}(\mathbf{x}_n) A_k, \quad (5.120)$$

where ϕ_{ik} is a diffusive flux of the form (5.95) with $\alpha = 1$. The interface states for the numerical flux are evaluated at the quadrature point directly by the polynomial representation of the solution over the spectral volume,

$$u_T(x, y) = \sum_{i \in \{C_T\}} u_i L_i(x, y), \quad (5.121)$$

and the neighbors. Note that we now store only the solution values for u . The gradient variables at the interface, which are needed to compute the consistent part of the numerical flux, are evaluated by differentiating the corresponding polynomial representations at each quadrature point. The two states are continuous over the faces inside the spectral volume and discontinuous over the SV faces. This means that the damping term of the numerical flux vanishes at quadrature points on the interior face, and plays its role at quadrature points on the SV faces. The length scale, L_r , in the numerical flux is determined similarly to the cell-centered schemes:

$$L_r = \frac{1}{2} |\Delta \mathbf{l}_{ik} \cdot \hat{\mathbf{n}}_{ik}|, \quad (5.122)$$

where $\Delta \mathbf{l}_{ik} = (x_k - x_i, y_k - y_i)$ and $\hat{\mathbf{n}}_{ik}$ is the unit outward normal vector of the k -th face. Again, it should be noted that the derived diffusion scheme has exactly the same implementation structure as a corresponding advection scheme: interface flux evaluated by two states meeting at a quadrature point.

To construct second-order diffusion schemes, we consider two strategies: the standard spectral-volume method and a cell-centered-type method. The resulting schemes are referred to as SV scheme and CCSV scheme.

SV Scheme: For second-order accuracy, we divide a triangular cell into three control volumes ($N = 3$) by connecting the face midpoints and the centroid (see Figure 5.8). Given three volume-averaged solutions, we can construct a polynomial of degree one ($m = 1$) in the form (5.121) over the spectral volume, T . Let $\{i_T\}$ denote a set of vertices of T . The shape function, $L_i(x, y)$, can be expressed in terms of the area-coordinates of T [71]:

$$L_i(x, y) = \frac{36}{5} Z_{i_T}(x, y) - \frac{7}{5}, \quad (5.123)$$

where $i_T \in \{i_T\}$ denotes the vertex of T belonging to the control volume i , and $Z_{i_T}(x, y)$ is the area-coordinate representing the area of a triangle defined by an arbitrary point (x, y) and the two vertices of T other than i_T , divided by the area of the triangle T . For example, for the control volume i in Figure 5.8, we have $i_T = 1$, and thus

$$Z_1(x, y) = \frac{x(y_2 - y_3) + x_2(y_3 - y) + x_3(y - y_2)}{x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)}, \quad (5.124)$$

where (x_1, y_1) , (x_2, y_2) , (x_3, y_3) are the vertex coordinates of T . It can be verified by direct integration that the polynomial thus defined satisfies the condition (5.115).

CCSV Scheme: Alternatively, we may construct a linear polynomial, u_T^{cc} , by interpolating the values, u_k , over the triangle (indicated by dotted lines in Figure 5.8) defined by the average coordinates of the vertices of the control volume:

$$u_T^{cc}(x, y) = \sum_{i \in \{C_T\}} u_i Z_i^{cc}(x, y), \quad (5.125)$$

where $Z_i^{cc}(x, y)$ is the area-coordinate defined over that triangle (not T). In this case, unlike $u_T(x, y)$, the polynomial u_T^{cc} passes through all values, u_k ; thus these values are considered as the solution values at the data points, not the volume-averaged values. This scheme is not a standard spectral-volume scheme. It may be considered as a cell-centered finite-volume scheme formulated on a spectral-volume mesh.

As in the discontinuous-Galerkin method, the damping term is crucial in the spectral-volume method also. That is, diffusion schemes may not be consistent without the damping term. The inconsistency problem has been pointed out in Ref. [18] for the one-dimensional diffusion equation. We will show later that the problem indeed occurs for both schemes considered here.

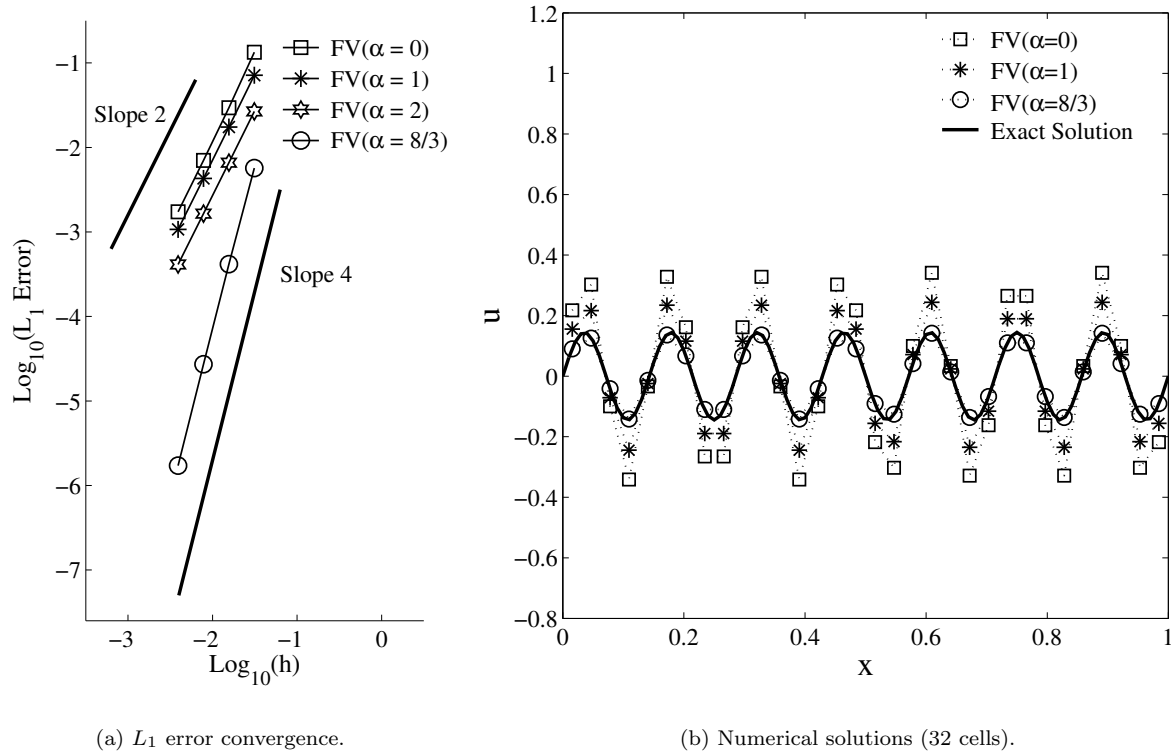


Figure 6.1. Numerical results for one-dimensional FV schemes.

6. Numerical Results

6.1. One-Dimensional Problem

We consider the following time-dependent diffusion problem:

$$u_t = \nu u_{xx}, \quad x \in (0, 1), \quad (6.1)$$

where $\nu = 1$, with the initial and boundary conditions,

$$u(x, 0) = \sin(14\pi x), \quad (6.2)$$

$$u(0, t) = 0, \quad (6.3)$$

$$u(1, t) = 0. \quad (6.4)$$

The exact solution is given by

$$u(x, t) = \exp(-196\pi^2 \nu t) \sin(14\pi x). \quad (6.5)$$

We compute the solution at time $t = 0.001$ on a series of uniform meshes: 32, 64, 128, and 256 cells. The initial solution was assigned in each cell by the exact cell-averaged value of Equation (6.2). The error at the final time is defined as a difference between the numerical solution and the exact cell-averaged value of Equation (6.5). For the SV schemes, the error is defined for the SV-averaged solution in each SV, not the CV-averaged solutions. The boundary conditions are specified by setting the solution in a ghost cell to be the negative of the solution in the adjacent interior cell, and enforcing equal gradients over the ghost and the adjacent interior cells. For all diffusion schemes, the time step is fixed as

$$\Delta t = 0.01 \Delta x^2. \quad (6.6)$$

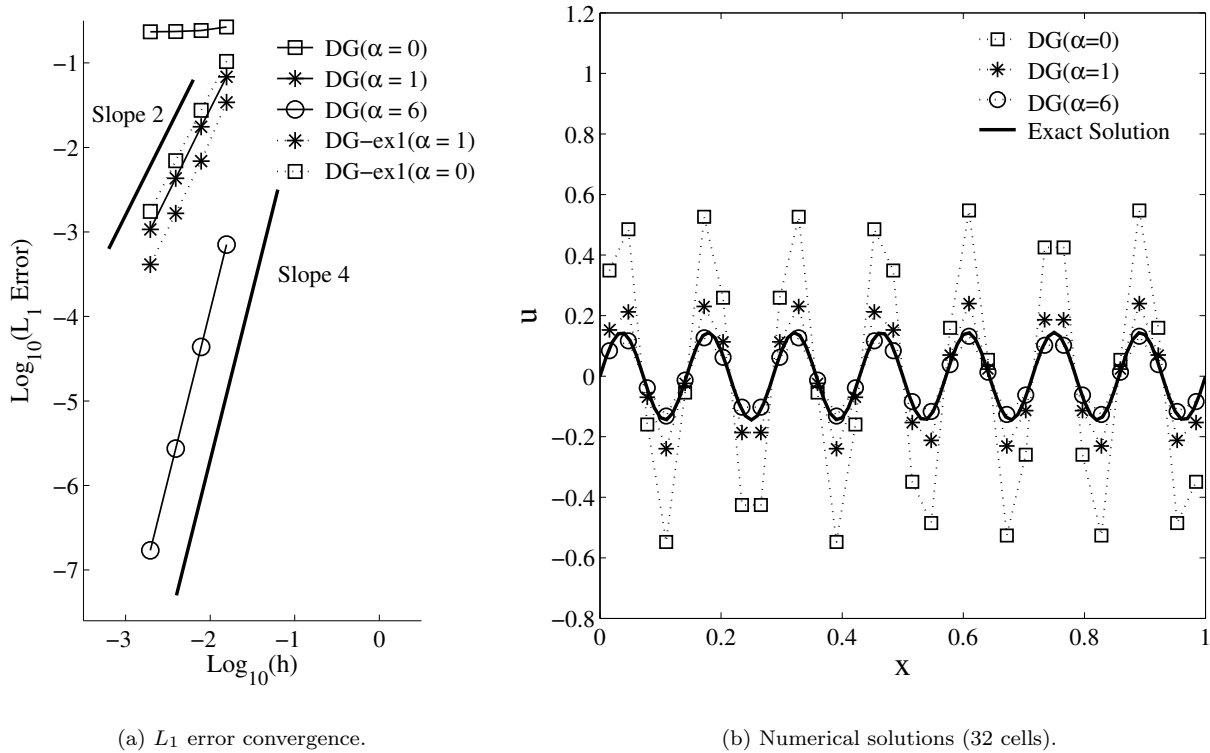


Figure 6.2. Numerical results for one-dimensional DG schemes.

Hence, all schemes take the same number of time steps to reach the final time: 104 steps for 32 cells, 411 steps for 64 cells, 1640 steps for 128 cells, and 6555 steps for 256 cells. For second-order schemes, we employ the forward-Euler time-stepping scheme to integrate them in time. For fourth-order schemes, we employ the classical fourth-order Runge-Kutta time integration scheme.

In Figure 6.1, the L_1 error convergence results and the solution plots (32 cells) are shown for the one-dimensional finite-volume (FV) diffusion schemes. The mesh size h in Figure 1(a) is a cell volume. As predicted, the FV diffusion scheme is second-order accurate with $\alpha = 1$, and fourth-order accurate with $\alpha = 8/3$. The FV scheme with $\alpha = 0$ is also second-order accurate, but slightly less accurate than the scheme with $\alpha = 1$. This is considered due to the lack of damping, which can be clearly seen in the solution plot. The scheme with $\alpha = 2$ corresponds to the standard three-point finite-difference scheme (the Galerkin scheme). The corresponding solution plot is not shown for clarity, but it lies between those of $\alpha = 1$ and $\alpha = 8/3$, as can be expected. The three-point scheme produces more accurate solutions than the other second-order schemes, but its maximum allowable time step is as small as the one for the fourth-order scheme ($\alpha = 8/3$).

In Figure 6.2, results are shown for the DG schemes constructed in Section 4.3. The mesh size h here is defined as half a cell volume to take into account the fact that the DG schemes carry two degrees of freedom per cell. The DG diffusion scheme is confirmed to be second-order accurate with $\alpha = 1$, fourth-order accurate with $\alpha = 6$, and inconsistent with $\alpha = 0$. As can be seen from the solution plot, the inconsistent scheme is highly inaccurate. It never approaches the exact solution in the grid refinement. Also shown in 2(a) are the error convergence results for the DG scheme derived as Example 1 in Section 4.3, which is here referred to as DG-ex1 scheme. It can be seen that this scheme is second-order accurate and produces more accurate solutions than the other DG scheme. Also, as predicted, this scheme retains second-order accuracy without the damping term ($\alpha = 0$) although the error is larger.

In Figure 6.3, results are shown for the SV schemes. Figure 3(a) confirms that the SV scheme is second-order accurate with $\alpha = 1$, fourth-order accurate with $\alpha = 3$, and inconsistent with $\alpha = 0$. Note that similarly to the DG case, the mesh size h is defined as half the cell volume. Again, we see that the inconsistent scheme gives highly inaccurate solution, and it does not converge to the exact solution with grid refinement. We note also that the SV scheme ($\alpha = 1$) gives more accurate solutions than the FV and DG schemes with $\alpha = 1$; it is

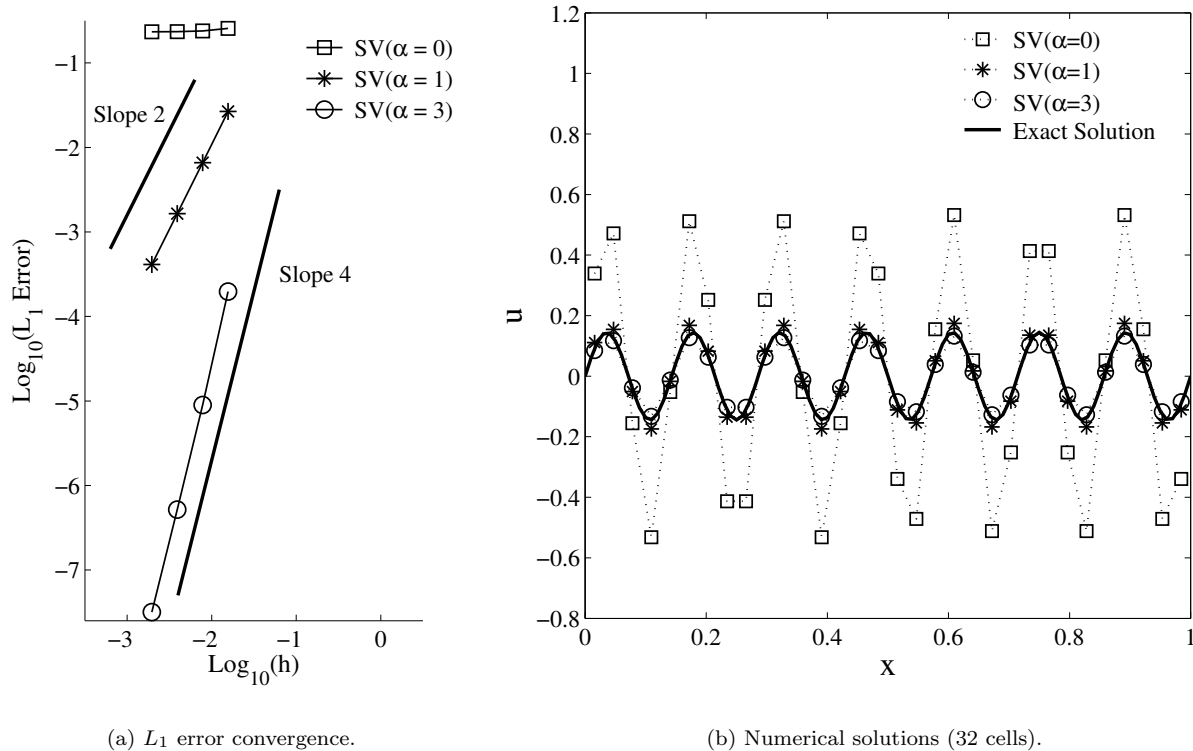


Figure 6.3. Numerical results for one-dimensional SV schemes.

FV($\alpha = 1$)	FV($\alpha = 2$)	FV($\alpha = 8/3$)	DG($\alpha = 1$)	DG($\alpha = 6$)	DG-ex1($\alpha = 1$)	SV($\alpha = 1$)	SV($\alpha = 3$)
$\frac{\Delta x^2}{\nu}$	$\frac{1}{2} \frac{\Delta x^2}{\nu}$	$\frac{3}{8} \frac{\Delta x^2}{\nu}$	$\frac{1}{4} \frac{\Delta x^2}{\nu}$	$\frac{1}{12} \frac{\Delta x^2}{\nu}$	$\frac{1}{2} \frac{\Delta x^2}{\nu}$	$\frac{1}{4} \frac{\Delta x^2}{\nu}$	$\frac{1}{12} \frac{\Delta x^2}{\nu}$

Table 6.1. Maximum explicit time step for the one-dimensional FV, DG, and SV diffusion schemes.

comparably accurate to the DG-ex1 scheme ($\alpha = 1$). We observed also from experiments that the SV scheme produced very similar second-order results for various other values of α such as 0.6, 1.5, 10, 100, etc.

Finally, the maximum allowable explicit (forward-Euler) time-steps are compared in Table 6.1, which are derived based on the Fourier analysis in Section 4. It shows that the FV schemes generally allow larger time steps than the DG and SV schemes considered here. Also, it can be seen that the fourth-order FV scheme allows as a large time step as the Galerkin scheme does, which is almost five times larger than the maximum time step for the fourth-order DG and SV schemes.

6.2. Two-Dimensional Problems

6.2.1. Isotropic Irregular Grids

We consider the following two-dimensional diffusion problem:

$$u_t = \nu(u_{xx} + u_{yy}), \quad (x, y) \in (0, 1) \times (0, 1), \tag{6.7}$$

where $\nu = 1$, with the initial solution,

$$u(x, y, 0) = 5 \sin(2\pi x) \sin(3\pi y), \tag{6.8}$$

and the boundary condition, $u = 0$ on the boundary. The exact solution to this problem is given by

$$u(x, y, t) = 5 \exp(-13\pi^2 \nu t) \sin(2\pi x) \sin(3\pi y). \tag{6.9}$$

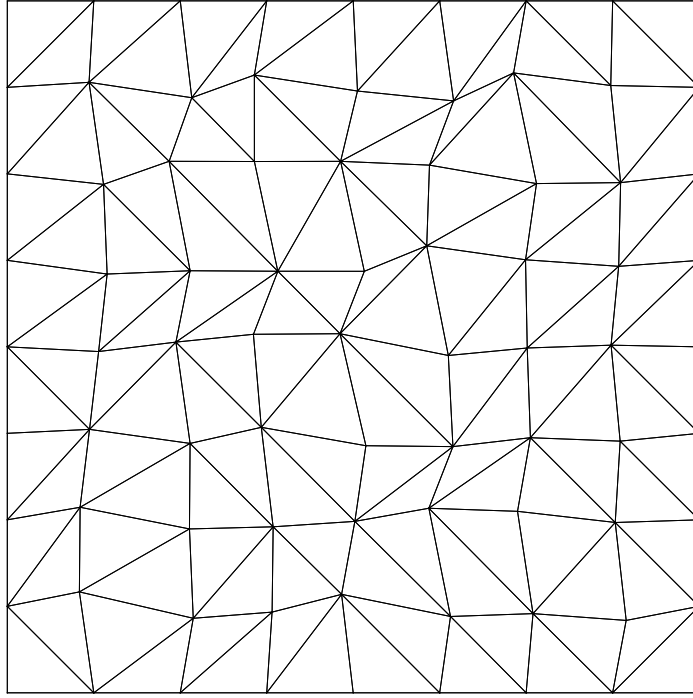


Figure 6.4. A typical isotropic irregular triangular grid (9×9 grid).

We compute the solution at time $t = 0.005$ on a series of irregular triangular grids. The irregular grids were generated by random diagonal splittings and nodal perturbations from uniform Cartesian grids of 9×9 , 17×17 , 33×33 , 65×65 nodes (see Figure 6.4). Although irregular, these grids have very low skewness. The L_1 -norm of the skewness measure, $\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}$, is 0.96 for all grids; the minimum value is 0.6. Boundary conditions are specified at the boundary nodes for node-based schemes, and at the cells having boundary faces for cell-based schemes (at control volumes for the SV schemes). All schemes are integrated in time by the forward-Euler time-stepping scheme with a fixed global time step:

$$\Delta t = 0.002 h^2, \quad (6.10)$$

where h is the minimum triangle height on a given grid. All schemes, therefore, take the same number of time steps to reach the final time: 320 steps for 9×9 grid, 1162 steps for 17×17 grid, 4893 steps for 33×33 grid, and 20,480 steps for 65×65 grid. For all schemes, the error is computed based on the difference between the exact solution and the numerical solution at data points: nodes for node-based schemes, centroids of triangles for cell-based schemes including DG and SV schemes.

For gradient reconstruction, we employ the Green-Gauss formula (5.27) for node-based finite-volume and residual-distribution schemes, except for the compact edge-based finite-volume scheme for which we employ the least-squares gradient as discussed in Section 5.3.3. For cell-centered finite-volume schemes, we employed the Green-Gauss gradient described in Section 5.5.1. In comparing the error convergence, we define the mesh size h as the square root of a control volume, except that it is defined as the square root of the 1/3 of the control volume for DG and SV schemes to take into account the fact that they carry three degrees of freedom per control volume.

In Figure 6.5, the results for the node-centered finite-volume schemes derived in Section 5.3.2 and the Galerkin scheme are shown. All schemes are confirmed to be second-order accurate. But the error for the edge-based finite-volume (EBFV) scheme with $\alpha = 0$ is significantly larger than others, demonstrating the lack of damping. We also find that the EBFV scheme with $\alpha = 1$ is comparably accurate to the Galerkin scheme and also to the edge-based Avg-LSQ-EN (EBFV-Avg-LSQ-EN) scheme. For the grids used, the EBFV-Avg-LSQ-FT scheme

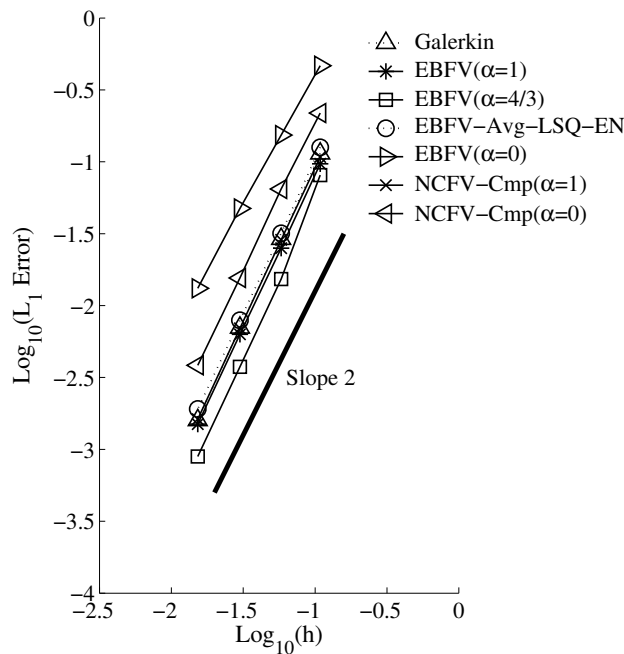


Figure 6.5. L_1 error convergence for node-centered finite-volume schemes on isotropic meshes.

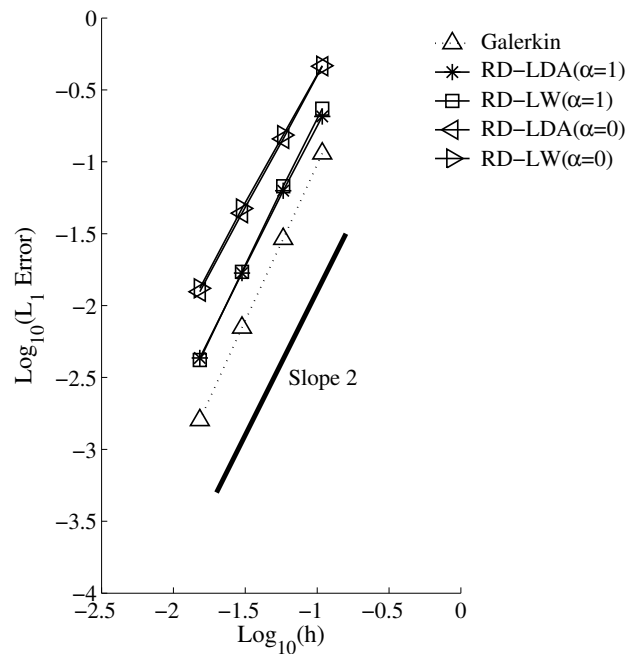


Figure 6.6. L_1 error convergence for RD schemes on isotropic meshes.

Galerkin	EBFV	EBFV($\alpha = 4/3$)	NCFV-Cmp	EBFV-Avg-LSQ-EN	RD-LDA	RD-LW
$0.3h^2$	$0.4h^2$	$0.3h^2$	$0.4h^2$	$0.6h^2$	$0.2h^2$	$0.6h^2$

Table 6.2. Maximum explicit time step experimentally determined for the FV and RD diffusion schemes for isotropic grids. $\alpha = 1$ unless otherwise stated.

is identical to the EBFV schemes with $\alpha = 1$. The node-centered compact finite-volume (NCFV-Cmp) scheme derived in Section 5.3.3 is also equally accurate, but it produces larger errors with $\alpha = 0$. We also tested the NCFV-Cmp scheme with the Green-Gauss gradients, which corresponds to the Braaten-Connell scheme [57] for triangular grids, and obtained very similar results to those obtained by using the least-squares gradients with $\alpha = 1$ (therefore, not shown). Finally, we see that the most accurate solution is obtained by the scheme with $\alpha = 4/3$.

Figure 6.6 shows the results for the residual-distribution schemes together with the Galerkin scheme for comparison. Both the LDA and Lax-Wendroff schemes are second-order accurate. They are slightly less accurate than the Galerkin scheme, but more accurate than those with $\alpha = 0$. That is, without damping, these RD schemes produce almost an order of magnitude larger errors than the Galerkin scheme.

Figure 6.7 shows the error convergence for the cell-centered finite-volume (CCFV) schemes in Section 5.5: the face-midpoint (CCFV-FM) scheme, the edge-midpoint (CCFV-EM) scheme, and the Avg-LSQ-EN scheme (CCFV-Avg-LSQ-EN). All schemes are confirmed to be second-order accurate. They are equally accurate except for the scheme with $\alpha = 0$. A lack of damping, again, leads to large solution errors. Using $\alpha = 4/3$, which gives fourth-order accuracy on structured grids, does not yield more accurate solutions here, but it gives much more accurate solutions for highly-skewed grids as will be shown later.

Figure 6.8 shows the error convergence for the DG schemes derived in Section 5.6, the Bassi-Rebay scheme [33], the LDG scheme [4], and the interior-penalty scheme of Shahbazi [47, 72]. Both the DG-I and DG-II schemes ($\alpha = 1$) are second-order accurate and comparably accurate with $\alpha = 1$. However, with $\alpha = 0$, the DG-I scheme turns unstable (thus results are not shown) and the DG-II scheme becomes inconsistent, which are both considered as due to a lack of damping. It is observed that the Bassi-Rebay and LDG schemes are both second-order accurate, but the errors are larger than those of the DG-I and DG-II schemes. Also plotted are the results for the DG-I and DG-II scheme with $\alpha = 3$ which corresponds to the choice for the fourth-order

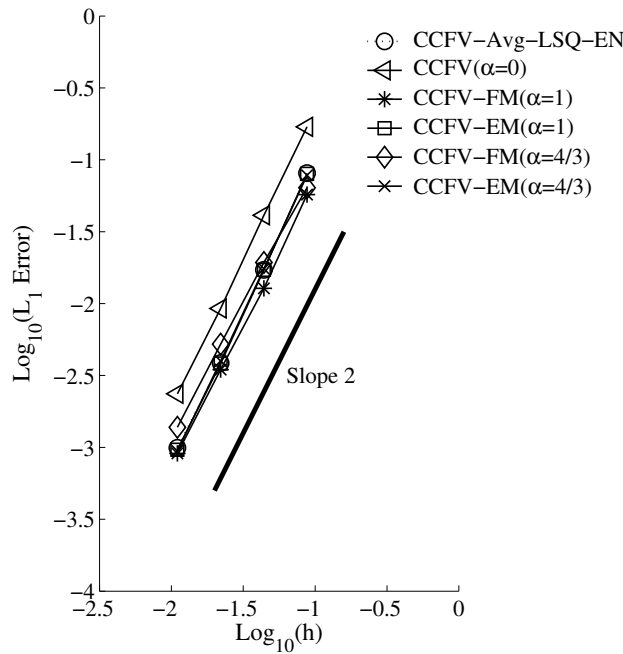


Figure 6.7. L_1 error convergence for cell-centered finite-volume schemes on isotropic meshes.

CCFV-FM	CCFV-EM	CCFV-Avg-LSQ-EN
$0.09h^2$	$0.08h^2$	$0.10h^2$

Table 6.3. Maximum explicit time step experimentally determined for the cell-centered FV diffusion schemes ($\alpha = 1$) for isotropic grids.

scheme in the one-dimensional DG scheme. Both schemes are second-order accurate, but the errors are larger than those with $\alpha = 1$ although still more accurate than the Bassi-Rebay scheme and equally accurate as the LDG scheme. For highly-skewed grids, however, these schemes produce highly accurate solutions as will be shown later. We also note that the interior-penalty scheme of Shahbazi gives larger errors than other schemes, which is considered due to its large damping coefficient. The L_1 norm of the corresponding α , i.e., Equation (5.100), was found to be 6.6 (min/max = 4.5/13.5) for the finest grid used in the experiment, and similarly for other grids.

Figure 6.9 shows the error convergence for the spectral-volume schemes derived in Section 5.7. The results for the Bassi-Rebay and LDG methods applied to the spectral-volume scheme (SV-Bassi-Rebay and SV-LDG schemes) are shown for comparison. Both the standard spectral-volume and cell-centered-type diffusion schemes are second-order accurate with $\alpha = 1$. The CCSV scheme gives slightly more accurate solutions. Also, both schemes are found to be inconsistent with $\alpha = 0$. We also tried $\alpha = 3/2$, which corresponds to the fourth-order one-dimensional scheme, but the results are very similar to those with $\alpha = 1$ (therefore, not shown).

Shown in Tables 6.2 - 6.5 are the maximum time steps allowable for some representative schemes. These maximum time steps were determined experimentally by running the test case with various time steps with a unit increment in the last digit in the number shown. We see from Table 6.2 that the Avg-LSQ-EN and RD-LW schemes allow relatively large time steps, the RD-LDA scheme corresponds to the smallest, and others are comparable. As shown in Table 6.3, all the cell-centered schemes are stable with comparable time steps. For the DG and SV schemes, Tables 6.4 and 6.5 show that all schemes are comparably stable except for the LDG scheme which requires an order of magnitude smaller time step. Note that the interior-penalty scheme of Shahbazi is under the most severe stability condition, which is considered due to its exceptionally large damping coefficient.

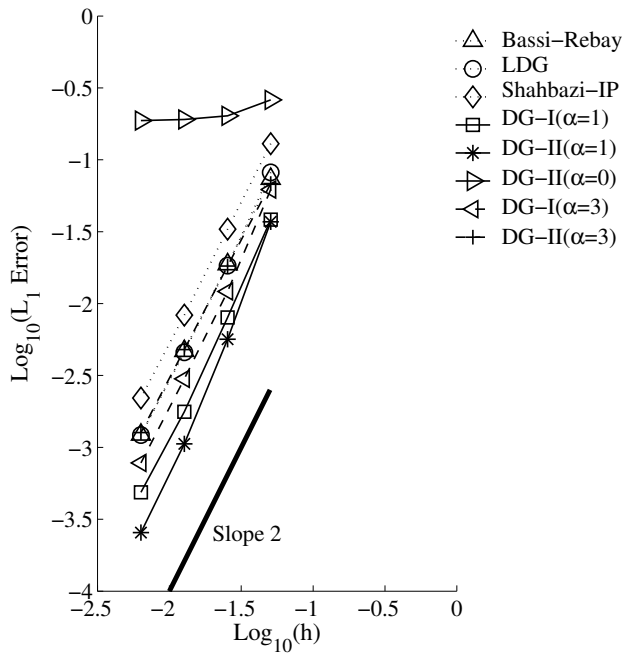


Figure 6.8. L_1 error convergence for DG schemes on isotropic meshes.

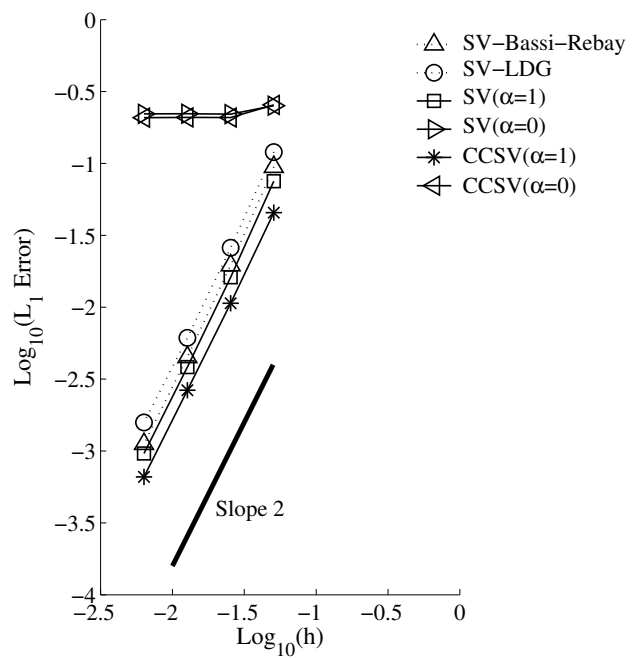


Figure 6.9. L_1 error convergence for SV schemes on isotropic meshes.

DG-I($\alpha = 1$)	DG-II($\alpha = 1$)	DG-I($\alpha = 3$)	DG-II($\alpha = 3$)	Bassi-Rebay	Shahbazi-IP	LDG
$0.02h^2$	$0.02h^2$	$0.008h^2$	$0.008h^2$	$0.03h^2$	$0.002h^2$	$0.004h^2$

Table 6.4. Maximum explicit time step experimentally determined for the DG diffusion schemes for isotropic grids.

SV($\alpha = 1$)	CCSV($\alpha = 1$)	SV-Bassi-Rebay	SV-LDG
$0.02h^2$	$0.02h^2$	$0.04h^2$	$0.01h^2$

Table 6.5. Maximum explicit time step experimentally determined for the SV diffusion schemes for isotropic grids.

6.2.2. Anisotropic Irregular Grids

We consider the following diffusion problem:

$$u_t = \nu(u_{xx} + u_{yy}), \quad (x, y) \in (0, 1) \times (0, 0.005), \quad (6.11)$$

where $\nu = 1$, with the initial solution,

$$u(x, y, 0) = 5 \sin(\pi x) \sin(4000\pi y), \quad (6.12)$$

and the boundary condition, $u = 0$ on the boundary. The exact solution to this problem is given by

$$u(x, y, t) = 5 \exp(-16000001\pi^2\nu t) \sin(\pi x) \sin(4000\pi y). \quad (6.13)$$

We compute the solution at time $t = 1.0E-08$ on a series of 15 irregular triangular grids. The irregular grids were generated by random diagonal splittings and nodal perturbations from uniform Cartesian grids of $n \times n$ nodes, where $n = 25, 33, 41, 49, 57, 65, 73, 81, 89, 97, 105, 113, 121, 129, 137$, resulting in the cell aspect-ratio of nearly 200 (see Figure 6.10). These grids are highly skewed. The L_1 -norm of the skewness measure, $\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}$,

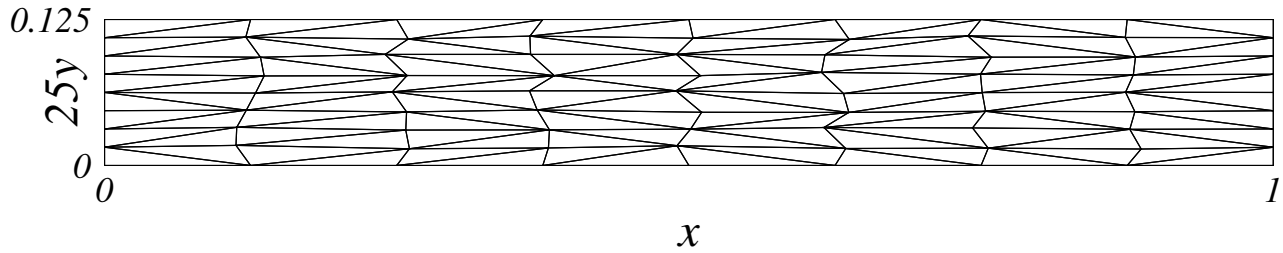


Figure 6.10. A typical high aspect-ratio irregular triangular grid (9x9 grid). The y-axis is stretched by a factor of 25 to show the details of the grid. Interior nodes are randomly perturbed in both coordinate directions.

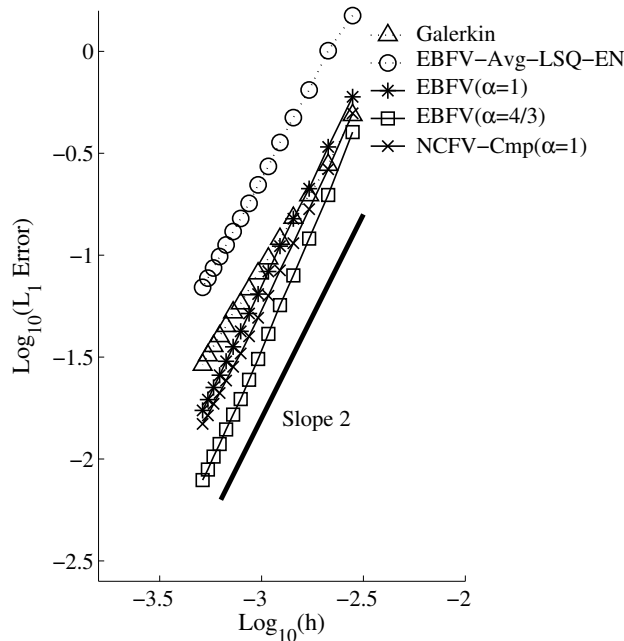


Figure 6.11. L_1 error convergence for node-centered finite-volume schemes on anisotropic meshes.

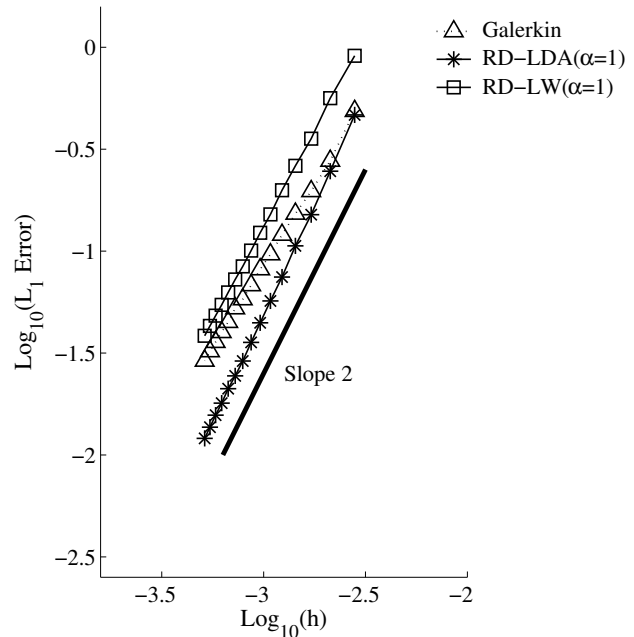


Figure 6.12. L_1 error convergence for RD schemes on anisotropic meshes.

was found to be 0.1 for all grids with the minimum value of around 0.007. All schemes are integrated in time by the forward-Euler time-stepping scheme with a fixed global time step:

$$\Delta t = 0.003 h^2. \tag{6.14}$$

Again, all schemes take exactly the same number of time steps to reach the final time: 77, 137, 214, 308, 419, 547, 692, 854, 1033, 1229, 1443, 1673, 1921, 2185, 2467, in the order of increasing grid size. For this test case, we do not show error convergence results for schemes with $\alpha = 0$ for clarity; these schemes perform poorly (inconsistent or unstable) as in the isotropic case.

Figure 6.11 shows the error convergence for the node-centered finite-volume schemes. It is observed that the EBFV and NCFV-Cmp schemes with $\alpha = 1$ or $\alpha = 4/3$ produce much more accurate solutions than the Galerkin scheme for this problem. Note that the Galerkin scheme is not quite second-order accurate. The Avg-LSQ-EN scheme, as expected, lacks damping for these highly-skewed grids, and resulted in significantly large errors. The edge-based scheme gives the most accurate solutions when $\alpha = 4/3$ as in the isotropic test case (slightly better than second-order accurate). We remark also that the NCFV-Cmp scheme with the Green-Gauss gradients (the Braaten-Connell scheme) gives very similar results to those obtained by using the least-squares gradients with $\alpha = 1$ (therefore, not shown).

Figure 6.12 shows the error convergence for the RD schemes. The Lax-Wendroff scheme is somewhat less accurate than the Galerkin scheme, but it gives a truly second-order convergence while the Galerkin scheme is not quite second-order accurate. On the other hand, the LDA scheme yields significantly most accurate solutions with a slightly higher convergence than second-order.

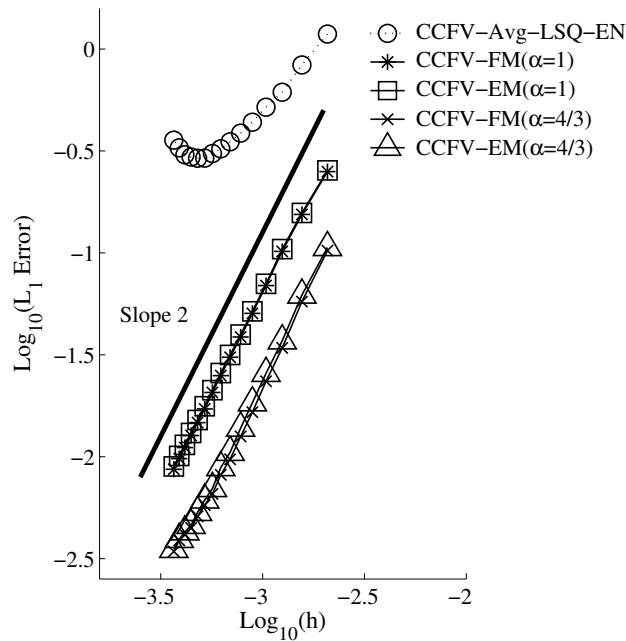


Figure 6.13. L_1 error convergence for cell-centered finite-volume schemes on anisotropic meshes, with unweighted-least-squares gradients.

Figure 6.13 shows the error convergence for the cell-centered finite-volume schemes. For these highly-skewed grids, we employed least-squares gradients, instead of the Green-Gauss gradients. The least-squares gradients were computed by the method described in Section 5.3.3 with no weights. It is a full reconstruction method that computes the gradient as well as the solution value at the cell center; the computed solution value is not needed, and thus ignored. We found that the diffusion schemes produced more accurate solutions with this method than with the widely-used method [63, 73, 74] which computes the gradient only (a least-squares problem with 2 unknowns). Also, for this anisotropic problem, we found that widely-used inverse-distance weighting led to larger errors in both gradients and solutions. As can be seen in Figure 6.13, the CCFV-AvgLSQ-EN scheme is, due to a lack of damping, significantly inaccurate, and gets unstable for fine grids. On the other hand, both the CCFV-FM and CCFV-EM schemes are second-order accurate with $\alpha = 1$ and $\alpha = 4/3$; the latter gives significantly more accurate solutions.

Figure 6.14 shows the error convergence for the DG schemes. All schemes are nearly second-order accurate. The Bassi-Rebay, LDG, DG-I($\alpha = 1$), and DG-II($\alpha = 1$) schemes are equally accurate. While the interior-penalty scheme of Shahbazi gives slightly more accurate solutions, the DG-I($\alpha = 3$), and DG-II($\alpha = 3$) schemes produce significantly more accurate solutions. These results indicate that a larger damping does not necessarily mean more accurate solutions. For the finest grid used in this experiment, the L_1 norm of the damping coefficient of Shahbazi's scheme, i.e., Equation (5.100), was found to be 48.8 (min/max = 4.0/869.7), and similarly for other grids.

Figure 6.15 shows the results for the SV schemes. Again, all schemes show nearly second-order behavior, and give comparably accurate solutions. Similarly to the isotropic case, the SV and CCSV schemes with $\alpha = 3$ yield similar solutions to those with $\alpha = 1$ (therefore, not shown).

Figures 6.16-6.43 show section plots of the numerical solutions at $y = 0.078125$ for various schemes on the 137x137 grid. The dash-dot curve is the section plot of the exact solution. Note that the plot of the exact solution is not perfectly smooth. This is because the section plot was created from the exact solution projected onto the computational grid that is irregular. Figure 6.16 shows that the Galerkin scheme produces highly oscillatory solution. This failure is considered due to the lack of positivity of the Galerkin scheme on grids that are not Delaunay [75]. A popular technique to enforce the positivity is to ignore all negative contributions (see Appendix E). We applied this technique to the Galerkin scheme and obtained the solution shown in Figure 6.17. Some high-frequency errors seem to have been eliminated, but the solution remains oscillatory. Note that as shown in Appendix E, with the positivity enforcement, the Galerkin scheme loses consistency: the error does not decrease with grid refinement. Figures 6.18 and 6.19 show the solutions obtained by the RD-LW with

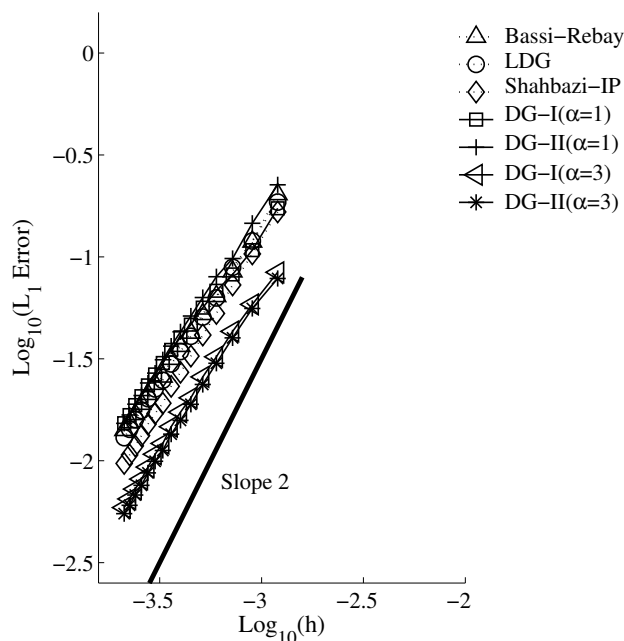


Figure 6.14. L_1 error convergence for DG schemes on anisotropic meshes.

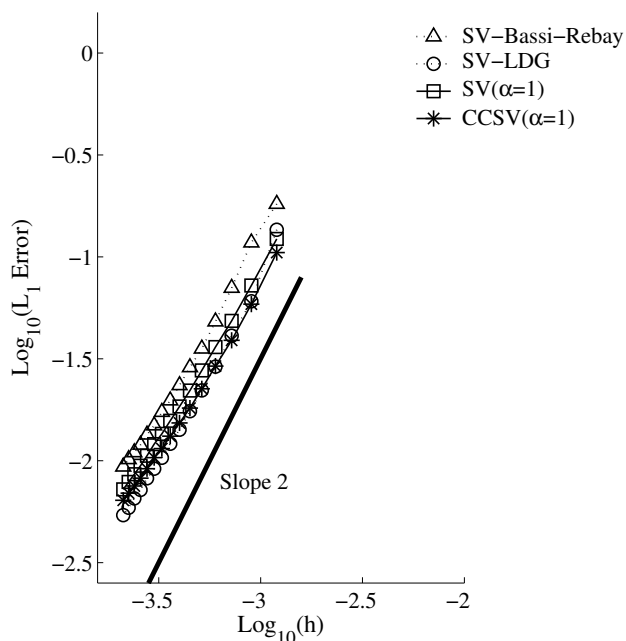


Figure 6.15. L_1 error convergence for SV schemes on anisotropic meshes.

$\alpha = 1$ and $\alpha = 8/3$, respectively. We see that the RD-LW gives oscillatory solutions similarly to the Galerkin scheme. Although it gives more accurate solution (on average) with $\alpha = 8/3$, but the solution is still oscillatory. A much more accurate solution is obtained by the RD-LDA scheme as in Figure 6.20. For comparison, the solution by the RD-LDA with $\alpha = 0$ is shown in Figure 6.21. Without the damping ($\alpha = 0$), the RD-LDA scheme gives highly oscillatory and inaccurate solution. Figures 6.22 and 6.23 show the results for the EBFV scheme with no damping ($\alpha = 0$) and the EBFV-AvgLSQ-EN scheme. Their solutions are similarly oscillatory and inaccurate, which demonstrates the lack of damping of the EBFV-AvgLSQ-EN scheme on highly-skewed grids. Very accurate solutions are obtained by the EBFV schemes as shown in Figures 6.24 and 6.25; the EBFV scheme with $\alpha = 4/3$ gives the best solution among all node-based schemes. Results for the NCFV-Cmp schemes are shown in Figures 6.26-6.29. As can be seen, the NCFV-Cmp schemes with the least-squares and Green-Gauss gradients, and with $\alpha = 1$ and $\alpha = 4/3$ give very similar results. These results show that the NCFV-Cmp schemes is insensitive to the gradient reconstruction methods and also to the choice of α . But the lack of damping does lead to a highly oscillatory solution (see Figure 6.29).

The rest of the results are for cell-based schemes. To create the section plots, we first compute nodal solutions by averaging the solution values extrapolated to the nodes from the surrounding triangles (using the gradients reconstructed or computed as unknowns in each cell), and then plot the solution in the same way as those of the node-based schemes. Figures 6.30 and 6.31 show the results for the CCFV-Avg-LSQ-EN scheme and the CCFV scheme with no damping ($\alpha = 0$). Both solutions are oscillatory and inaccurate as expected, again, demonstrating the lack of damping of the Avg-LSQ-EN scheme on highly-skewed grids. On the other hand, the CCFV-FM schemes with the least-squares gradients are much more accurate (those for the CCFV-EM schemes are similar), especially with $\alpha = 4/3$, as shown in Figures 6.32 and 6.33. Shown in Figures 6.34-6.41 are the results for DG schemes. The Bassi-Rebay scheme and the LDG schemes give somewhat oscillatory solutions but not too far from the exact solution (see Figures 6.34 and 6.35). The solutions obtained by the DG-I and DG-II schemes are more oscillatory with $\alpha = 1$ (Figures 6.36 and 6.37), and catastrophically inaccurate with $\alpha = 0$ (Figure 6.39). The interior-penalty scheme of Shahbazi gives a better but still somewhat oscillatory solution as shown Figure 6.38. The solutions obtained with $\alpha = 3$ are shown in Figures 6.40 and 6.41. Again, although still somewhat oscillatory, the solutions are accurate on average. Very similar results between Figures 6.38 (Shahbazi-IP: $L_1(\alpha) = 48.8$) and 6.41 ($\alpha = 3$) imply that the solution quality is not significantly affected by the amount of damping. For SV schemes, the SV-Bassi-Rebay scheme produces a slightly irregular solution although accurate on average as shown in Figures 6.42 while the SV-LDG scheme produces a much smoother solution to this problem as shown in Figure 6.43. The SV and CCSV schemes with $\alpha = 1$ give slightly oscillatory

but, on average, accurate solutions, as shown in Figures 6.44 and 6.45. Finally, the SV and CCSV schemes with $\alpha = 0$ lead to, due to the lack of damping, extremely inaccurate solutions.

It should be noted that oscillatory solutions may become smooth, through a number of time steps (relaxations), in the steady state. These results do not necessarily imply similar behaviors for steady state problems. Also, a more sophisticated, perhaps implicit, time-integration scheme may provide additional damping to combat the oscillations. If that is the case, those shown here to give oscillatory but averagely accurate solutions may turn out to be good candidates for practical applications.

7. Concluding Remarks

“So, what is the best diffusion scheme?”

“I give you a recipe. You find the best one. Taste is quite subjective, you know?”

7.1. General Remarks

In this paper, we have introduced a general principle for constructing diffusion schemes: discretize the first-order hyperbolic diffusion system by an advection scheme and then derive a diffusion scheme from it by discarding extra equations and explicitly evaluating the solution gradient. We have demonstrated the principle by deriving various second-order diffusion schemes for node/cell-centered finite-volume, residual-distribution, discontinuous-Galerkin, and spectral-volume methods for uniform grids in one dimension and for unstructured grids in two dimensions. It has been shown that a damping term, which is important for high-frequency error damping, is automatically incorporated into the derived diffusion scheme through the dissipation term of the generating advection scheme for all methods considered. A parameter α , which represents the ratio of the maximum explicit time step to the relaxation time of the hyperbolic diffusion system, has been shown to play a role of controlling the damping effect of the derived diffusion scheme. Special values of α have been discovered for one-dimensional schemes that make them fourth-order accurate, and they have been shown to give significantly accurate solutions even in two dimensions. For two-dimensional schemes, a skewness parameter has been shown to be incorporated automatically into the damping coefficient, enabling robust and accurate computations on highly-skewed grids. Numerical results have been shown to demonstrate the design accuracy of the derived diffusion schemes and the importance of the damping term (a lack of damping leads to unstable and/or inconsistent results). Comparison with widely-used schemes have been made to demonstrate that these derived diffusion schemes give comparable or more accurate solutions for time-dependent diffusion problems on a uniform grid in one dimension, and on isotropic/anisotropic irregular triangular grids in two dimensions. These results show particularly for the DG and SV methods that the derived diffusion schemes are more efficient than those based on a first-order system since they directly act on the interface values just like advection schemes and no extra variables (and equations) are required.

7.2. Derived Diffusion Schemes

Diffusion schemes derived from the proposed principle are summarized below.

1. Finite-Volume Schemes

- (a) In one dimension, a one-parameter family of second-order diffusion schemes has been derived. It becomes fourth-order accurate with $\alpha = 8/3$.
- (b) A variety of two-dimensional node-centered diffusion schemes have been constructed: edge-based, compact, element-based schemes. The edge-based scheme has been shown to produce significantly accurate solutions with $\alpha = 4/3$.
- (c) Two-dimensional cell-centered diffusion schemes have been constructed. The choice $\alpha = 4/3$ gives significantly accurate solutions for highly-skewed grids.
- (d) Widely-used average-least-squares schemes have been shown to be a subset of the edge-based/cell-centered diffusion schemes derived in this paper.
- (e) A widely-used edge-normal average-least-squares scheme has been shown to break down for highly-skewed grids due to a lack of damping, while the derived diffusion schemes produce very accurate results on such grids due to the damping term that is automatically amplified on such grids.

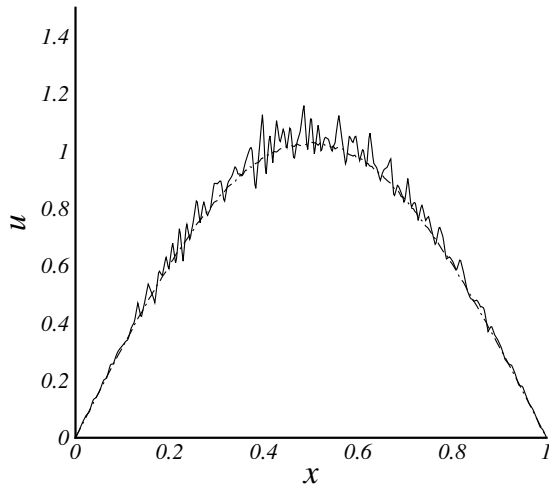


Figure 6.16. Section plot for Galerkin scheme.

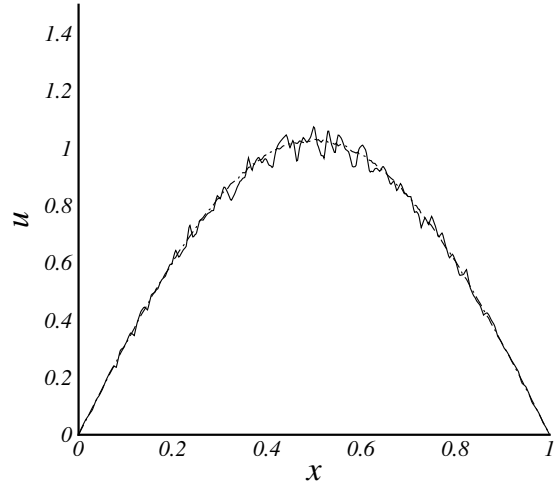


Figure 6.17. Section plot for Galerkin scheme with positivity enforcement.

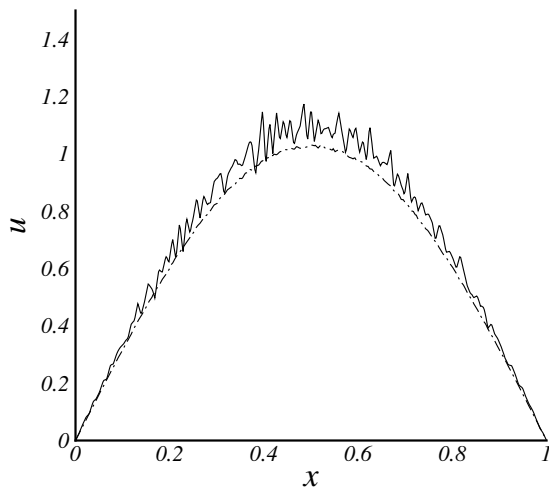


Figure 6.18. Section plot for RD-LW scheme ($\alpha = 1$).

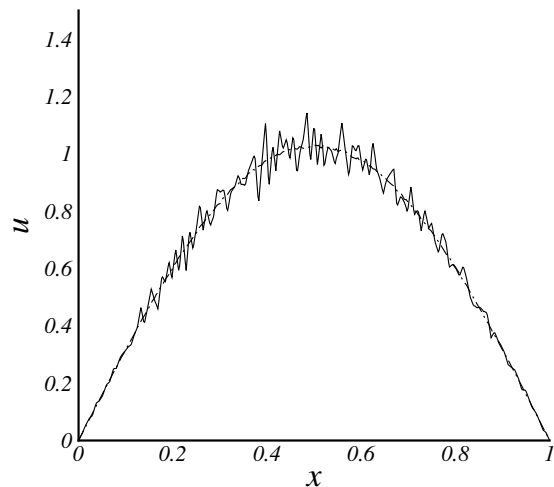


Figure 6.19. Section plot for RD-LW scheme ($\alpha = 8/3$).

2. Residual-Distribution Schemes

- (a) In one dimension, a one-parameter family of second-order diffusion scheme has been constructed. It has been shown to be equivalent to the finite-volume diffusion scheme.
- (b) In two dimensions, diffusion schemes have been constructed from the LDA and Lax-Wendroff schemes, and a damping term is identified for each scheme.
- (c) The LDA diffusion scheme has been shown to produce very accurate solutions on highly-skewed grids.

3. Discontinuous Galerkin Schemes

- (a) A one-parameter family of one-dimensional second-order diffusion schemes has been derived, which has been shown to be fourth-order accurate with $\alpha = 6$.
- (b) A two-dimensional diffusive flux has been constructed from the upwind flux. It has been shown to

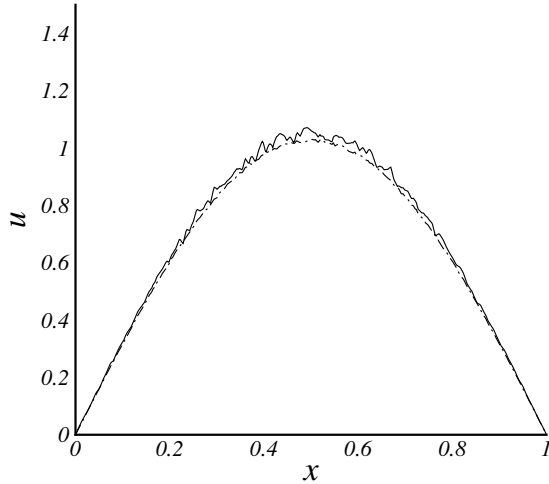


Figure 6.20. Section plot for RD-LDA scheme ($\alpha = 1$).

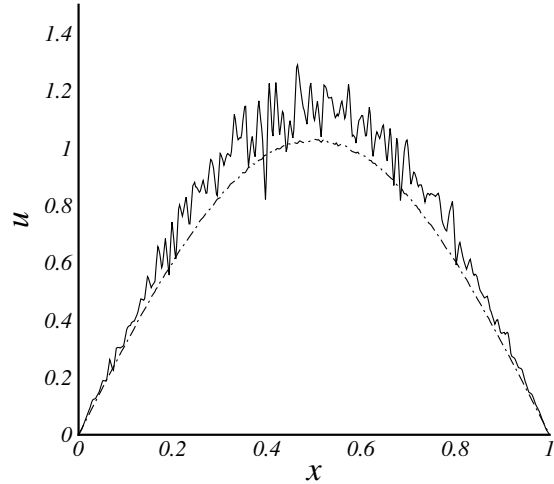


Figure 6.21. Section plot for RD-LDA scheme ($\alpha = 0$).

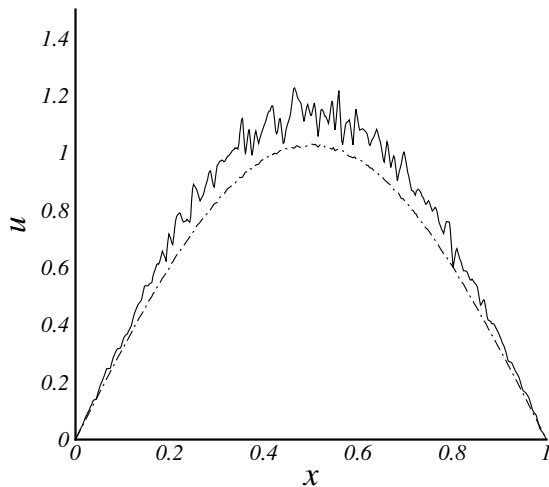


Figure 6.22. Section plot for Avg-LSQ-EN scheme.

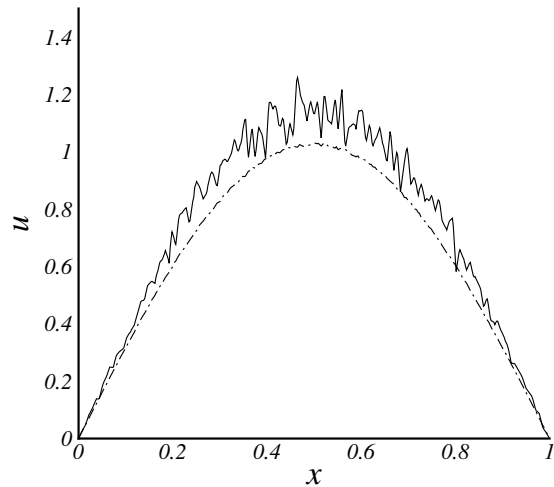


Figure 6.23. Section plot for EBFV scheme ($\alpha = 0$).

give comparably or more accurate results than the Bassi-Rebay and LDG schemes, and also than Shahbazi's interior penalty scheme.

- (c) Two different methods for the associated volume integral have been considered. Both methods have been shown to give satisfactory and comparable results with $\alpha = 1$ as well as $\alpha = 3$.
- (d) The choice $\alpha = 3$ gives more accurate results than $\alpha = 1$ for highly-skewed grids.

4. Spectral-Volume Schemes

- (a) A one-parameter family of one-dimensional second-order diffusion schemes has been derived, which has been shown to be fourth-order accurate with $\alpha = 3$.
- (b) Two-dimensional diffusion schemes have been constructed based on two different polynomial constructions. Both schemes have been shown to give comparably accurate results with those with the Bassi-Rebay and LDG schemes.

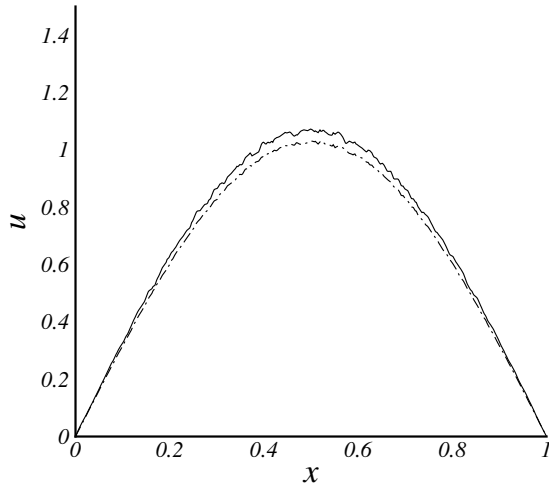


Figure 6.24. Section plot for EBFV scheme ($\alpha = 1$).

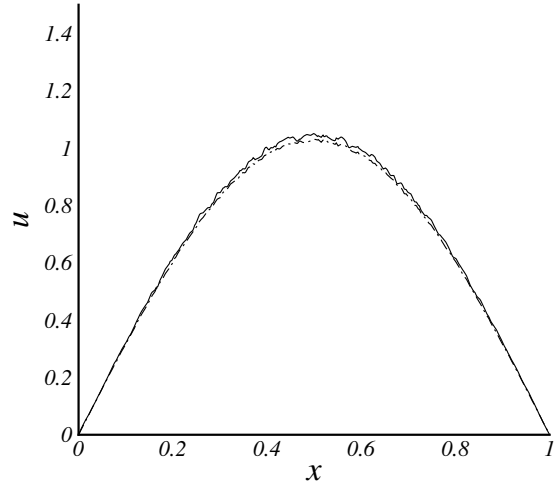


Figure 6.25. Section plot for EBFV scheme ($\alpha = \frac{4}{3}$).

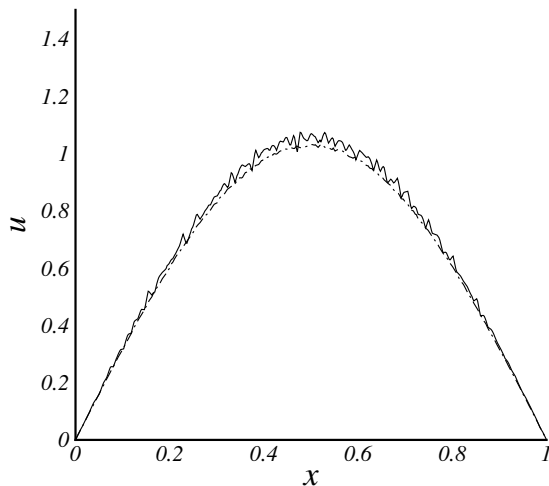


Figure 6.26. Section plot for NCFV-Cmp scheme ($\alpha = 1$).

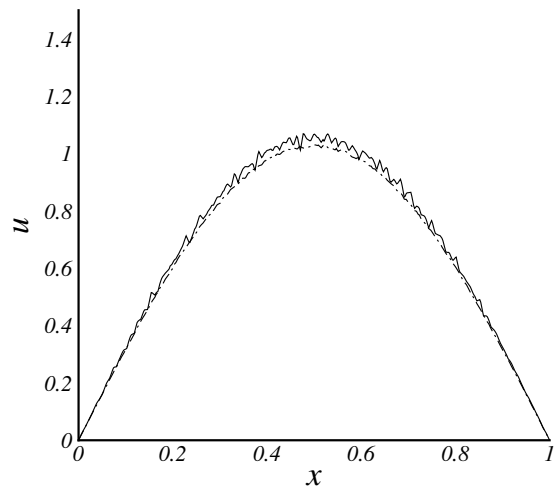


Figure 6.27. Section plot for NCFV-Cmp scheme with the Green-Gauss gradients (the Braaten-Connell scheme).

- (c) Spectral-volume diffusion schemes are insensitive to the value of α . The choice $\alpha = 3/2$, which corresponds to the value that achieves fourth-order accuracy in one dimension, gives similar results for $\alpha = 1$.

7.3. Advection-Diffusion Problems

Many diffusion schemes derived from the proposed principle can be implemented in the framework of advection schemes. For the finite-volume, discontinuous-Galerkin, and spectral-volume methods, the schemes are implemented based on the interface flux acting on two discontinuous states. This feature will enable a very simple and efficient integration of a diffusion scheme with a corresponding advection scheme for advection-diffusion problems. It is particularly useful for specialized advection schemes such as the multidimensional reconstruction scheme of Gnoffo [76] where the diffusive flux derived in this paper can be computed together with the advective flux by the same interface states. For the residual-distribution method, the LDA and Lax-Wendroff

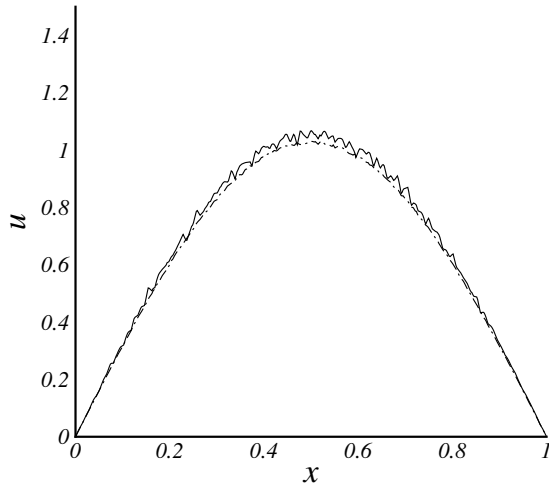


Figure 6.28. Section plot for NCFV-Cmp scheme ($\alpha = \frac{1}{3}$).

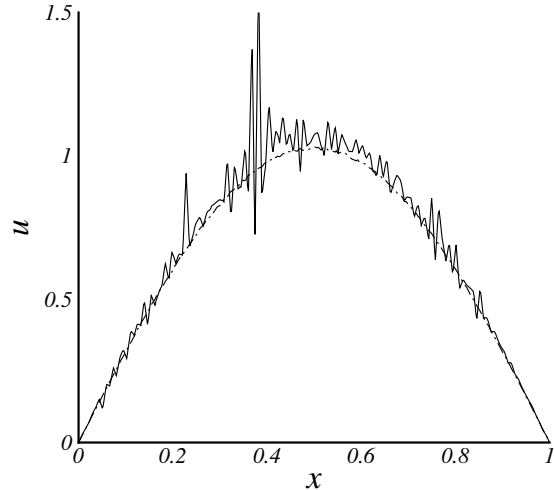


Figure 6.29. Section plot for NCFV-Cmp scheme ($\alpha = 0$).

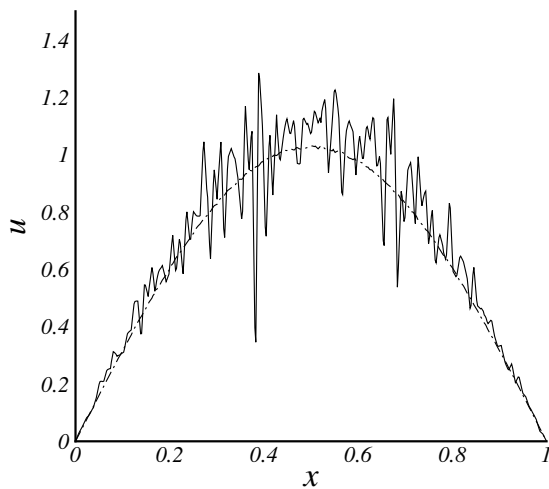


Figure 6.30. Section plot for CCFV-Avg-LSQ-EN.

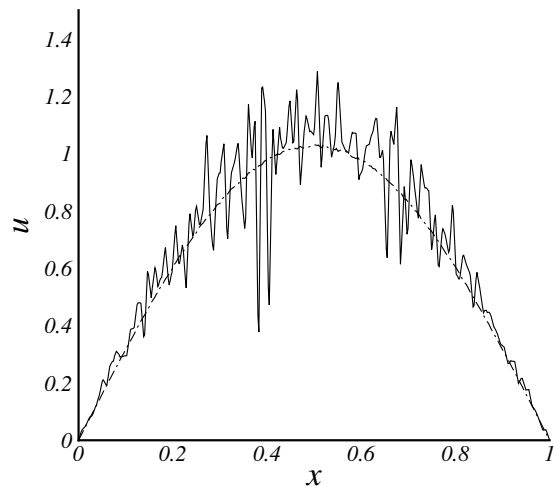


Figure 6.31. Section plot for CCFV-FM scheme ($\alpha = 0$).

diffusion schemes can be combined with the LDA, Lax-Wendroff, or other advection scheme by defining a single distribution coefficient as a Reynolds-number weighted sum of the two coefficients [9, 34], distributing a sum of advective and diffusive cell-residuals. The distribution of the cell-residual for the entire advection-diffusion equation is very important to preserve the accuracy of the scheme for all Reynolds numbers [26, 77]. We remark also that to construct an advection-diffusion scheme, it is possible to apply the proposed principle directly to the first-order advection-diffusion system introduced in Refs. [9, 34]. A benefit of this approach (particularly for the residual-distribution method) is that an advection scheme and a diffusion scheme are automatically integrated; no special consideration is necessary to combine the two.

7.4. Future Work

Future work includes extensions to yet other discretization methods. The general principle introduced in this paper is independent of the discretization method, and it should be easily applied to many other methods such

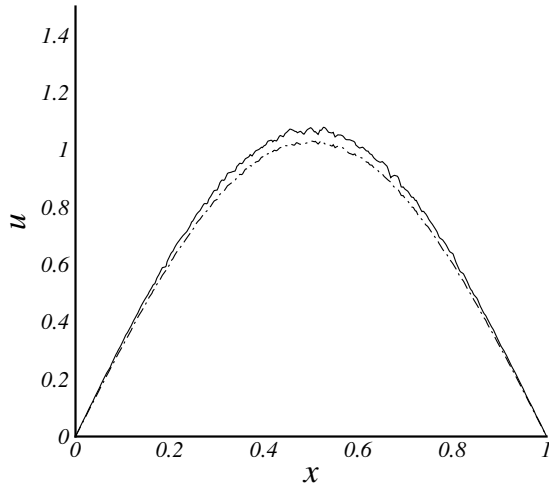


Figure 6.32. Section plot for CCFV-FM scheme ($\alpha = 1$).

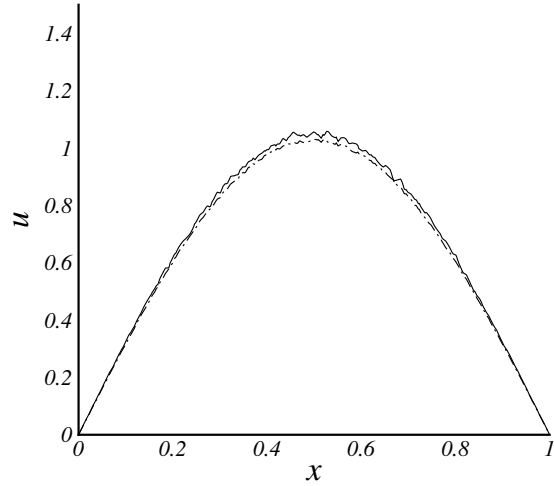


Figure 6.33. Section plot for CCFV-FM scheme ($\alpha = \frac{1}{3}$).

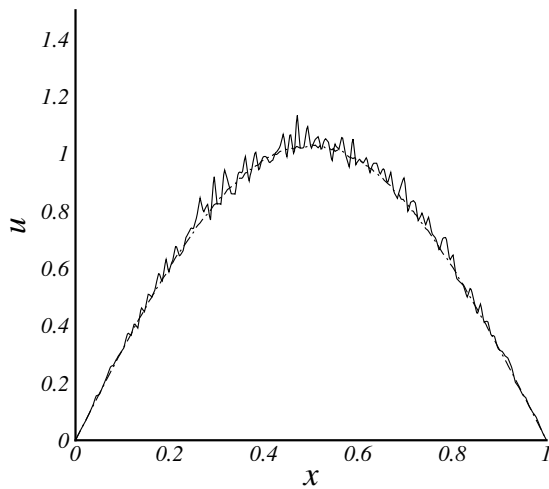


Figure 6.34. Section plot for Bassi-Rebay scheme.

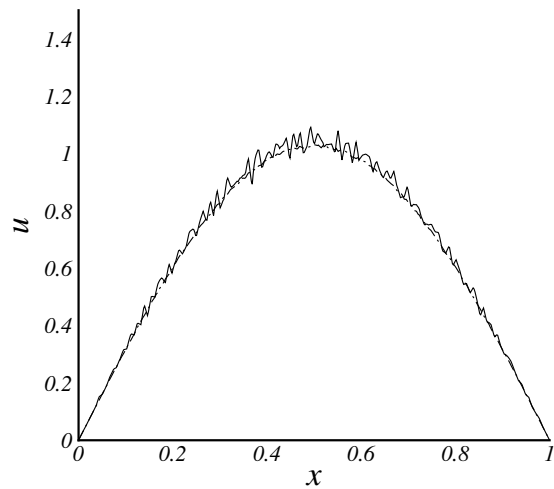


Figure 6.35. Section plot for LDG scheme.

as the cell-vertex method [78], the spectral-difference method [79], the compact finite-difference method [80], the meshless method [81], etc. Extensions to three dimensions are straightforward. Many diffusion schemes derived in this paper are directly applicable to three-dimensional grids. If that is not the case, a diffusion scheme can always be derived, provided an appropriate three-dimensional advection scheme is available: discretize the three-dimensional first-order hyperbolic diffusion system and extract a diffusion scheme from the result. For practical applications, implicit formulations of the schemes derived in this paper should be explored. The positivity of the damping coefficient is expected to improve diagonal dominance of the resulting linear system for highly-skewed grids. Also, the performance of the diffusion schemes for steady state computations should be studied, not just for accuracy but also for convergence to the steady state, perhaps in multigrid methods where high-frequency damping is essential for grid-independent convergence. In extensions to advection-diffusion problems, the impact of limiters, which are typically incorporated into advection schemes, on the diffusion schemes should be studied. Extensions to higher-order schemes should also be explored. Higher-order diffusion schemes may be derived simply from higher-order advection schemes: follow the principle. For the residual-distribution method,

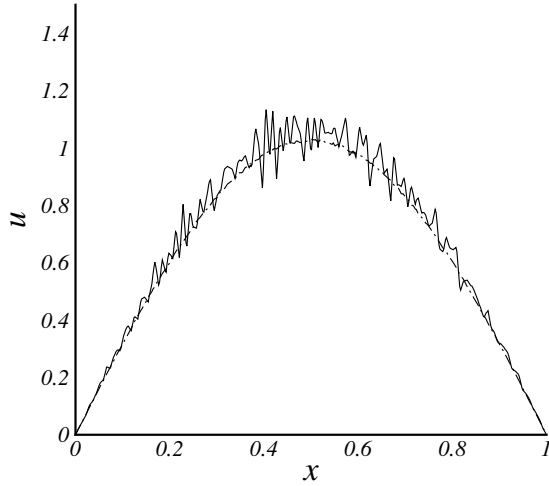


Figure 6.36. Section plot for DG-I scheme ($\alpha = 1$).

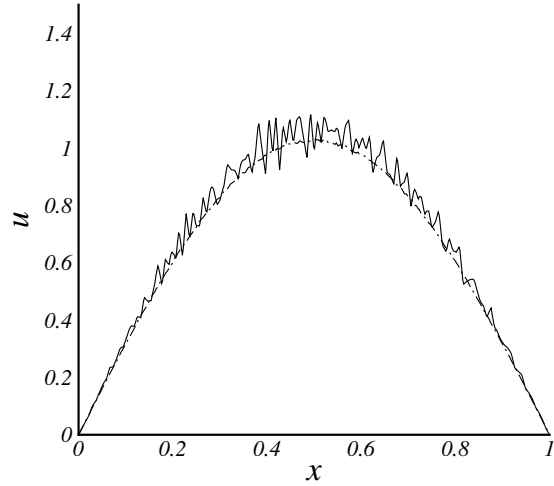


Figure 6.37. Section plot for DG-II scheme ($\alpha = 1$).

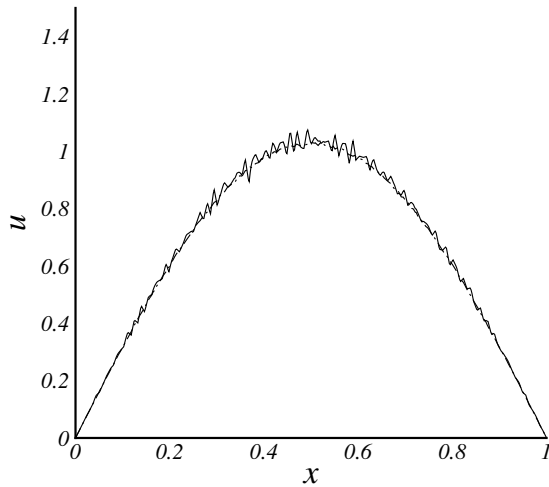


Figure 6.38. Section plot for Shahbazi-IP scheme.

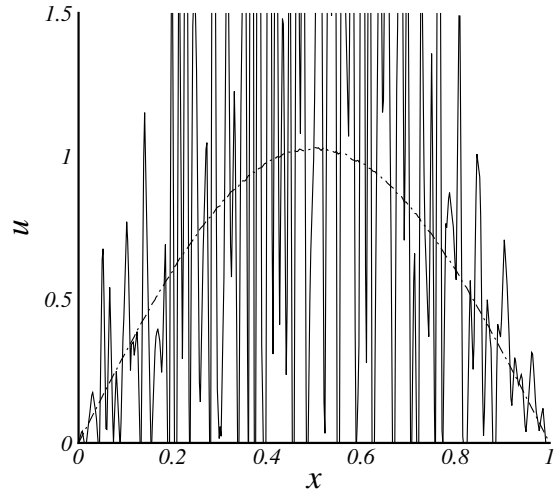


Figure 6.39. Section plot for DG-II scheme ($\alpha = 0$).

compact high-order diffusion schemes may be devised based on high-order elements [77,82]: evaluate the gradient variables by direct differentiation of the high-order polynomial in which the damping term is expected to play an important role. For applications to nonlinear systems, such as the Navier-Stokes equations, there are at least two strategies. One is to go back to the common approach and regard the derived diffusion scheme as a means to obtain the gradient on the control-volume boundary. For example, the diffusive flux for the edge-based scheme (5.25) can be written in the form:

$$\phi_{jk} = -\nu \nabla u|_{jk} \cdot \hat{\mathbf{n}}_{jk}, \quad (7.1)$$

where $\nabla u|_{jk}$, which is considered as the gradient approximation on the control-volume boundary, is given by

$$\nabla u|_{jk} = \frac{1}{2} [(\nabla u)_k + (\nabla u)_j] + \frac{\alpha}{2L_r} (u_R - u_L) \hat{\mathbf{n}}_{jk}. \quad (7.2)$$

This formula can be employed to evaluate the gradients at the control-volume boundary by which the diffusive flux is then directly computed. The other is to follow the principle and derive a diffusion scheme from an

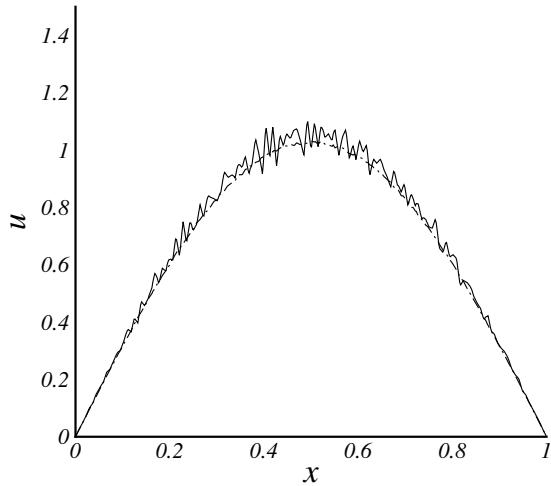


Figure 6.40. Section plot for DG-I scheme ($\alpha = 3$).

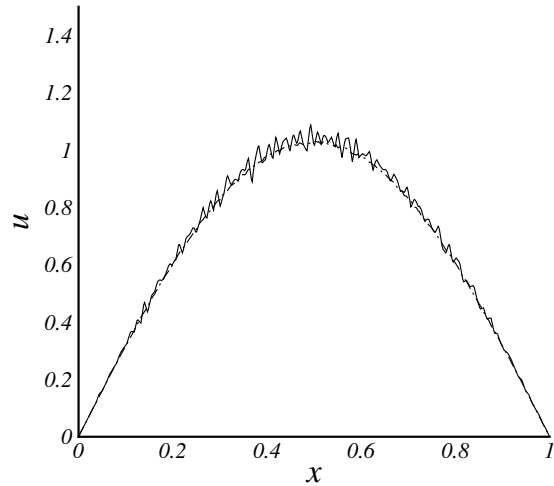


Figure 6.41. Section plot for DG-II scheme ($\alpha = 3$).

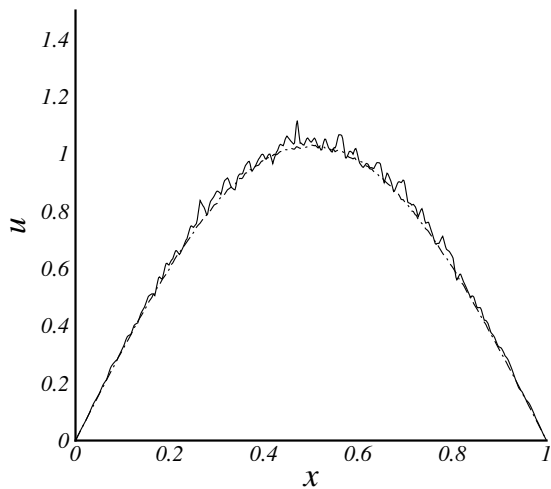


Figure 6.42. Section plot for SV-Bassi-Rebay scheme.

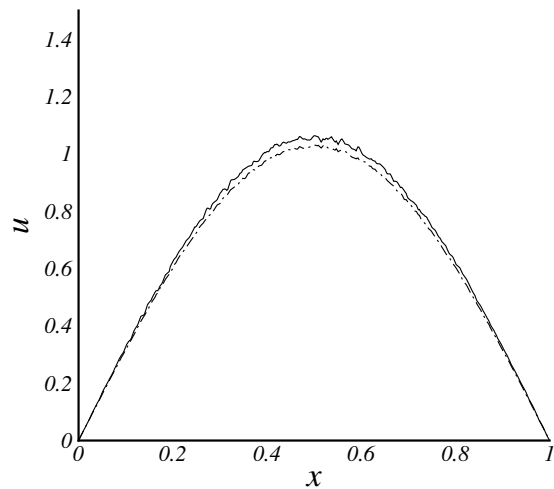


Figure 6.43. Section plot for SV-LDG scheme.

advection scheme. The latter would be a more sensible option if well-established discretization techniques for nonlinear hyperbolic systems could be directly applied.

Finally, the principle introduced in this paper, being based on advection schemes, has a great potential to expand the variety of diffusion schemes as much as we have for advection schemes. It may be worth exploring other options by using different advection schemes from those employed here, especially for nonlinear diffusion equations.

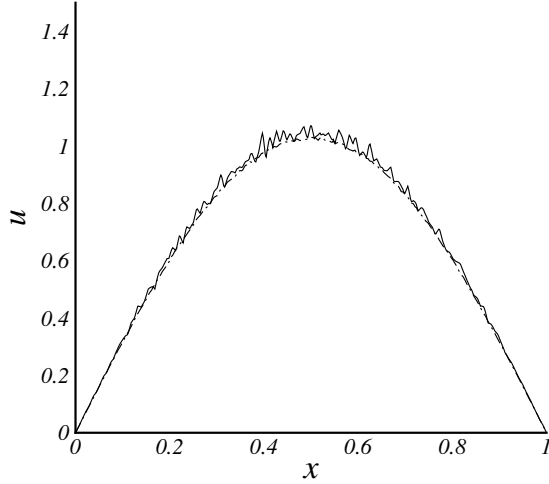


Figure 6.44. Section plot for SV scheme ($\alpha = 1$).

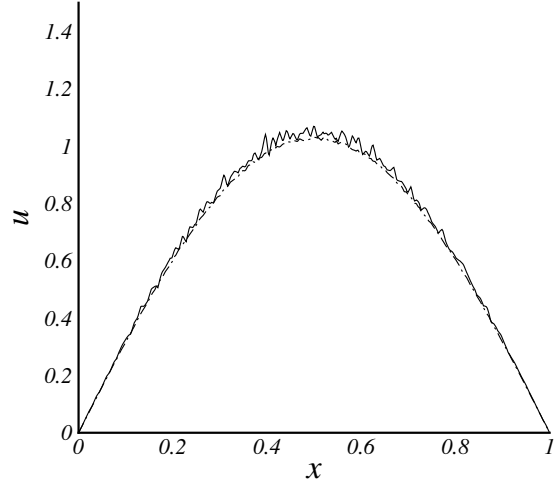


Figure 6.45. Section plot for CCSV scheme ($\alpha = 1$).

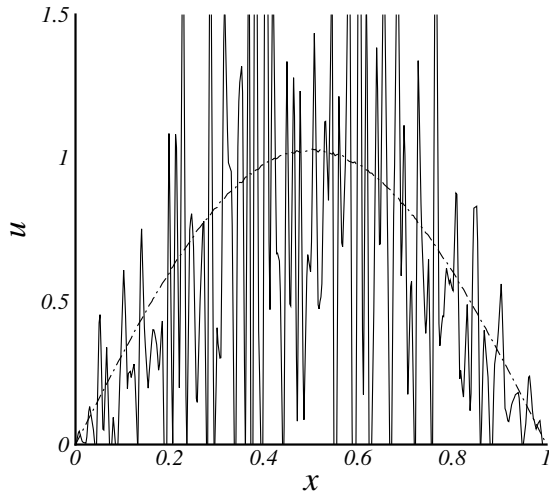


Figure 6.46. Section plot for SV scheme ($\alpha = 0$).

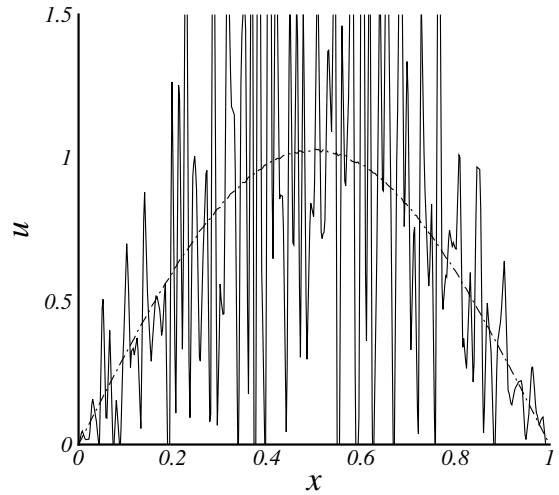


Figure 6.47. Section plot for CCSV scheme ($\alpha = 0$).

Appendix A. Bassi-Rebay and Penalized Bassi-Rebay Scheme

We derive the Bassi-Rebay diffusion scheme from the DG discretization of the first-order diffusion system. We begin with the weak formulation (5.85):

$$\int_{T_j} v \frac{\partial \mathbf{U}}{\partial t} dV = - \sum_{k \in \{K_j\}} \int_{\partial T_j^k} v \mathbf{H} dA_k + \int_{T_j} \nabla v \cdot (\mathbf{F}, \mathbf{G}) dV + \int_{T_j} v \mathbf{Q} dV. \quad (\text{A.1})$$

Discretizing the surface integral by an N -point quadrature per face,

$$\int_{T_j} v \frac{\partial \mathbf{U}}{\partial t} dV = - \sum_{k \in \{K_j\}} \sum_{n=1}^N w_{k_n} v_{k_n} \Phi_{k_n} A_k + \int_{T_j} \nabla v \cdot (\mathbf{F}, \mathbf{G}) dV + \int_{T_j} v \mathbf{Q} dV. \quad (\text{A.2})$$

where w_{k_n} are the quadrature weights, v_{k_n} is the basis function, and Φ_{k_n} is a numerical flux evaluated at the quadrature point \mathbf{x}_{k_n} :

$$v_{k_n} = v(\mathbf{x}_{k_n}), \quad \Phi_{k_n} = \begin{bmatrix} \phi_{k_n} \\ \mathbf{h}_{k_n} \end{bmatrix} = \Phi_{k_n}(\mathbf{x}_{k_n}). \quad (\text{A.3})$$

For the sake of convenience, we write the equation for u separately, with the gradient variables denoted as a vector by $\boldsymbol{\tau}$, i.e., $\boldsymbol{\tau} = [p, q]^t$,

$$\int_{T_j} v \frac{\partial u}{\partial t} dV = - \sum_{k \in \{K_j\}} \sum_{n=1}^N w_{k_n} v_{k_n} \phi_{k_n} A_k - \nu \int_{T_j} \nabla v \cdot \boldsymbol{\tau} dV, \quad (\text{A.4})$$

$$\int_{T_j} v \frac{\partial \boldsymbol{\tau}}{\partial t} dV = - \sum_{k \in \{K_j\}} \sum_{n=1}^N w_{k_n} v_{k_n} \mathbf{h}_{k_n} A_k - \frac{1}{T_r} \int_{T_j} u \nabla v dV - \frac{1}{T_r} \int_{T_j} v \boldsymbol{\tau} dV. \quad (\text{A.5})$$

In Section 5.6, we ignored the second equation and set $\boldsymbol{\tau} = \nabla u$ in each element to derive a diffusion scheme. But here we consider determining $\boldsymbol{\tau}$ by forcing the right hand side of the second equation to vanish:

$$\int_{T_j} v \boldsymbol{\tau} dV = - \sum_{k \in \{K_j\}} \sum_{n=1}^N w_{k_n} v_{k_n} (T_r \mathbf{h}_{k_n}) A_k - \int_{T_j} u \nabla v dV. \quad (\text{A.6})$$

This equation is solved for $\boldsymbol{\tau}$, which is then substituted into Equation (A.4) to construct a diffusion scheme. The resulting DG diffusion scheme differs, therefore, from the one in Section 5.6 in that it evaluates the gradient variables by solving Equation (A.6), not by differentiating u . As mentioned in Section 3.1, Equation (A.6) may result in a global system of equations for $\boldsymbol{\tau}$. For example, if we employ the upwind flux for the numerical flux Φ_{k_n} ,

$$\Phi_{k_n} = \frac{1}{2} [\mathbf{H}(\mathbf{U}_{k_n}^+) + \mathbf{H}(\mathbf{U}_{k_n}^-)] - \frac{1}{2} |\mathbf{A}_n| (\mathbf{U}_{k_n}^+ - \mathbf{U}_{k_n}^-). \quad (\text{A.7})$$

then,

$$\phi_{k_n} = -\frac{\nu}{2} [\boldsymbol{\tau}_{k_n}^+ + \boldsymbol{\tau}_{k_n}^-] \cdot \hat{\mathbf{n}}_k - \frac{1}{2} \sqrt{\frac{\nu}{T_r}} (u_{k_n}^+ - u_{k_n}^-), \quad (\text{A.8})$$

$$T_r \mathbf{h}_{k_n} = -\frac{1}{2} (u_{k_n}^+ + u_{k_n}^-) \hat{\mathbf{n}}_k - \frac{1}{2} \sqrt{T_r \nu} [(\boldsymbol{\tau}_{k_n}^+ - \boldsymbol{\tau}_{k_n}^-) \cdot \hat{\mathbf{n}}_k] \hat{\mathbf{n}}_k. \quad (\text{A.9})$$

In this case, because of the second term in Equation (A.9), Equation (A.6) becomes a globally coupled sparse linear system for $\boldsymbol{\tau}$. On the other hand, if we employ the central flux, i.e., the flux (A.7) without the dissipation term, then, $\boldsymbol{\tau}$ can be determined locally, and the result is the Bassi-Rebay scheme:

$$\int_{T_j} v \frac{\partial u}{\partial t} dV = - \sum_{k \in \{K_j\}} \sum_{n=1}^N w_{k_n} v_{k_n} \phi_{k_n} A_k - \nu \int_{T_j} \nabla v \cdot \boldsymbol{\tau} dV, \quad (\text{A.10})$$

$$\int_{T_j} v \boldsymbol{\tau} dV = - \sum_{k \in \{K_j\}} \sum_{n=1}^N w_{k_n} v_{k_n} \mathbf{h}_{k_n}^* A_k - \int_{T_j} u \nabla v dV, \quad (\text{A.11})$$

where

$$\phi_{k_n} = -\frac{\nu}{2} [\boldsymbol{\tau}_{k_n}^+ + \boldsymbol{\tau}_{k_n}^-] \cdot \hat{\mathbf{n}}_k, \quad \mathbf{h}_{k_n}^* = -\frac{1}{2} (u_{k_n}^+ + u_{k_n}^-) \hat{\mathbf{n}}_k. \quad (\text{A.12})$$

It is also possible to replace ϕ_{k_n} in Equation (A.12) by Equation (A.8) to include the damping (or penalty) term for ϕ_{k_n} . Then, $\boldsymbol{\tau}$ can still be determined locally. The resulting scheme may be called the Penalized-BR scheme. This approach can be employed for the spectral-volume method also. Such a spectral-volume scheme may be called the SV-Penalized-BR scheme. A very similar spectral-volume scheme is discussed in Ref. [3], which is different from the above scheme only in the definition of the coefficient for the damping term (the above scheme has the skewness measure in the denominator).

These Penalized-BR schemes were tested for the highly-skewed grid problem in Section 6.2.2. The results are shown in Figures A.1 and A.2. For the DG method, the Penalized-BR scheme improves the accuracy of the Bassi-Rebay scheme. For the SV method, it only gives slight improvement over the SV-Bassi-Rebay scheme.

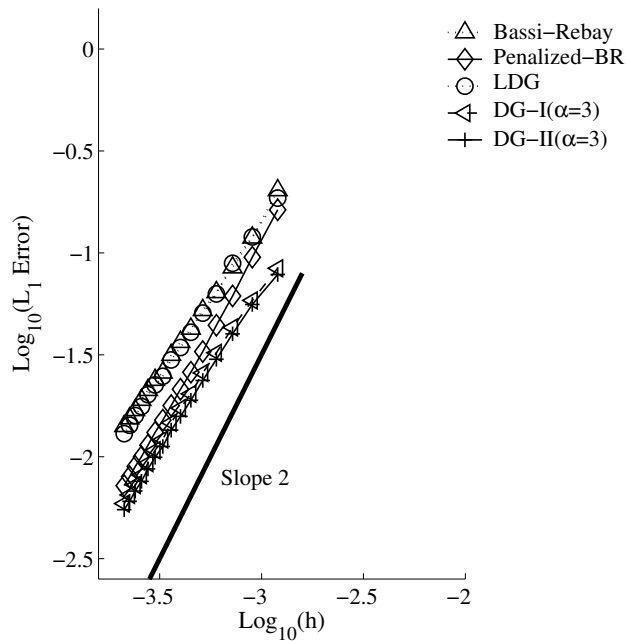


Figure A.1. L_1 error convergence for DG schemes on anisotropic meshes, including the Penalized-BR scheme.

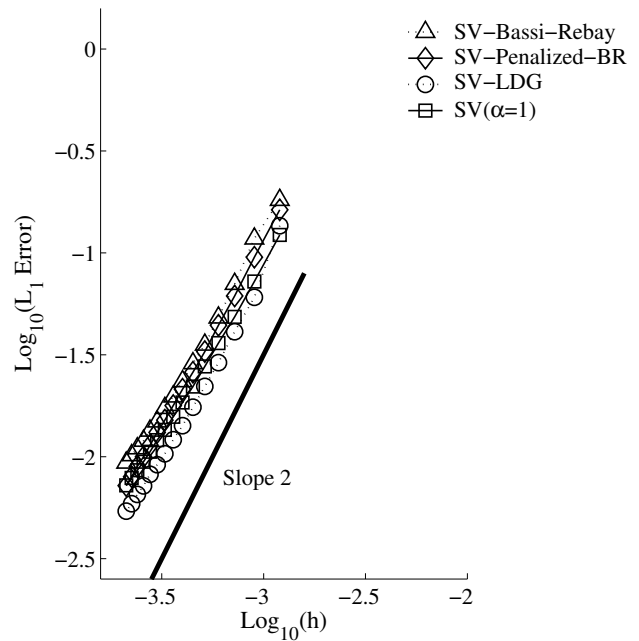


Figure A.2. L_1 error convergence for SV schemes on anisotropic meshes, including the SV-Penalized-BR scheme.

Appendix B. Linearity-Preserving Finite-Volume Integration Formulas

The edge-based integration formula commonly employed for edge-based finite-volume schemes is known to be exact for linear functions (linearity-preserving) for pure triangular and tetrahedral grids [75], but not necessarily exact on other types of grids [20, 52]. We present a simple derivation of finite-volume integration formulas for triangles, quadrilaterals, tetrahedra, hexahedra, prisms, and pyramids, to provide a clear explanation of the dependence of the exactness on the type of grids. Particular boundary weights to ensure the exactness of the formulas will also be derived. Some of these weights have been known for a long time, but a simple proof does not seem available in the literature. All results are presented for the integral of the gradient; therefore the results can be used to compute the gradient that is exact for linear functions. Each component can be directly applied to the integral of the flux divergence (i.e., conservation laws) as a quadrature that is exact for linear fluxes.

B.1. Two Dimensions

B.1.1. Triangle

Consider a triangular cell, T , in Figure B.1, whose boundary will be denoted by ∂T , the outward normal by \mathbf{n} , and the boundary length by A . We derive a finite-volume integration formula over the dual control volume around the node 1 (the shaded area) for the integral:

$$\int_{dual} \text{grad } u \, dV. \quad (\text{B.1})$$

Note that the volume of the dual, V_{dual} , is precisely $\frac{1}{3}$ of the volume of the cell, V :

$$V_{dual} = \frac{1}{3}V. \quad (\text{B.2})$$

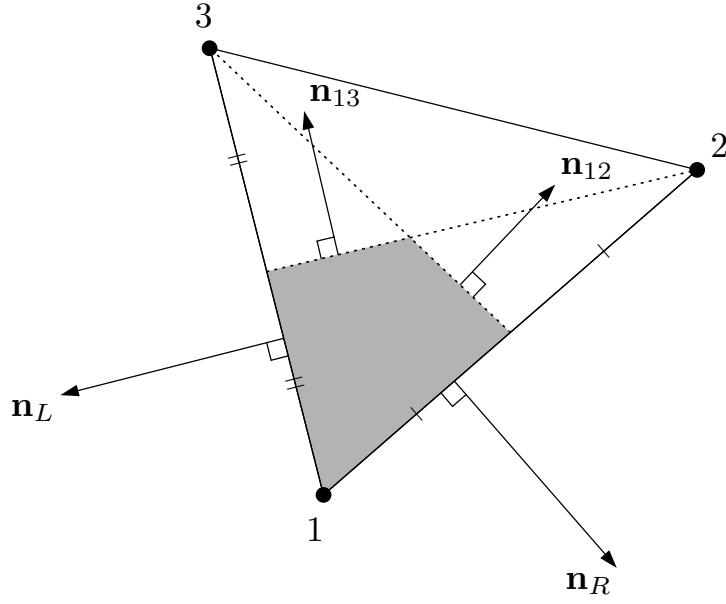


Figure B.1. Dual control volume around a node 1, and scaled outward normals.

Assume now that u is a linear function, so that the gradient is a global constant that can be exactly expressed by the Green-Gauss gradient,

$$\text{grad } u = \frac{1}{V} \int_T \text{grad } u \, dV \quad (\text{B.3})$$

$$= \frac{1}{V} \int_{\partial T} u \, \mathbf{n} \, dA \quad (\text{B.4})$$

$$= \frac{1}{V} \left(\frac{u_1 + u_2}{2} \begin{bmatrix} y_2 - y_1 \\ x_1 - x_2 \end{bmatrix} + \frac{u_2 + u_3}{2} \begin{bmatrix} y_3 - y_2 \\ x_2 - x_3 \end{bmatrix} + \frac{u_3 + u_1}{2} \begin{bmatrix} y_1 - y_3 \\ x_3 - x_1 \end{bmatrix} \right) \quad (\text{B.5})$$

$$= \frac{1}{V} \left(\frac{u_1}{2} \begin{bmatrix} y_2 - y_3 \\ -(x_2 - x_3) \end{bmatrix} + \frac{u_2}{2} \begin{bmatrix} y_3 - y_1 \\ -(x_3 - x_1) \end{bmatrix} + \frac{u_3}{2} \begin{bmatrix} y_1 - y_2 \\ -(x_1 - x_2) \end{bmatrix} \right), \quad (\text{B.6})$$

which is thus written as

$$\text{grad } u = \frac{1}{2V} \sum_{i=1}^3 u_i \, \mathbf{n}_i, \quad (\text{B.7})$$

where u_i is the function value at the node i , and \mathbf{n}_i is the scaled inward normal of the edge opposite to the node i . Then, the integral (B.1) can be evaluated as follows:

$$\int_{\text{dual}} \text{grad } u \, dV = (\text{grad } u) \times \frac{1}{3} V = \frac{1}{6} (u_1 \, \mathbf{n}_1 + u_2 \, \mathbf{n}_2 + u_3 \, \mathbf{n}_3). \quad (\text{B.8})$$

Substituting into Equation (B.8) the following geometric identities,

$$\mathbf{n}_1 = 3(\mathbf{n}_{12} + \mathbf{n}_{13}) + 5(\mathbf{n}_L + \mathbf{n}_R), \quad (\text{B.9})$$

$$\mathbf{n}_2 = 3\mathbf{n}_{12} + \mathbf{n}_R, \quad (\text{B.10})$$

$$\mathbf{n}_3 = 3\mathbf{n}_{13} + \mathbf{n}_L, \quad (\text{B.11})$$

where \mathbf{n}_{12} , \mathbf{n}_{13} , \mathbf{n}_L , \mathbf{n}_R are scaled outward normals of the dual volume as indicated in Figure B.1, we arrive at the finite-volume integration formula:

$$\int_{\text{dual}} \text{grad } u \, dV = \frac{u_1 + u_3}{2} \mathbf{n}_{13} + \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \left(\frac{5}{6} u_1 + \frac{1}{6} u_3 \right) \mathbf{n}_L + \left(\frac{5}{6} u_1 + \frac{1}{6} u_2 \right) \mathbf{n}_R. \quad (\text{B.12})$$

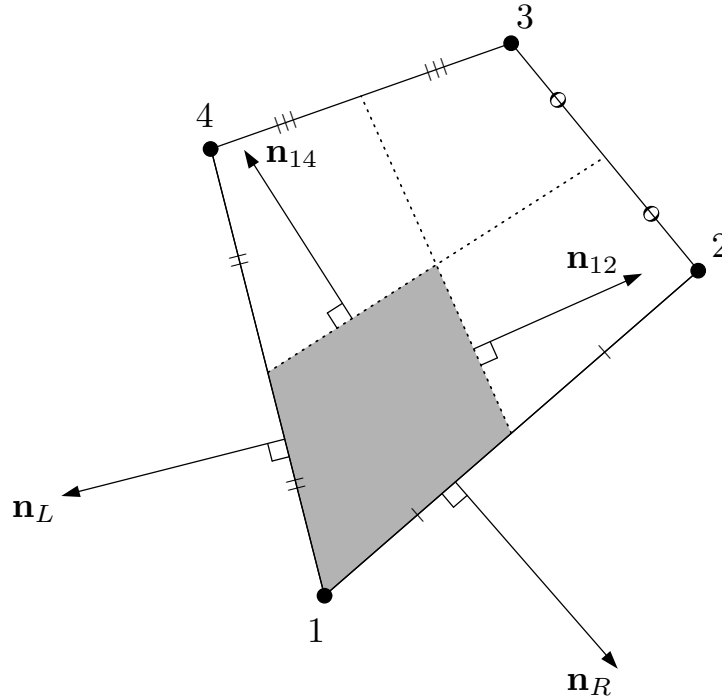


Figure B.2. Quadrilateral cell, a median dual volume around a node 1, and scaled outward normals.

This formula is, therefore, exact for any linear function, u . Note that the boundary contributions over the edges, $\{1, 2\}$ and $\{1, 3\}$, are evaluated both with the same set of weights, $5/6$ and $1/6$, without reference to the third node. These weights have long been known [53, 20], but this simple derivation does not seem available in the literature.

B.1.2. Quadrilateral - Median Dual

Consider a general quadrilateral cell shown in Figure B.2. We assume that u is a linear function, so that the gradient is given by the Green-Gauss gradient over the triangle $\{1, 2, 4\}$. Then, by multiplying the Green-Gauss gradient by the median dual volume (the shaded area), we obtain the following finite-volume integration formula:

$$\int_{dual} \text{grad } u \, dV = \left[\frac{1}{4} + \frac{1}{16} \left(\frac{V_3}{V_1} - 1 \right) \right] (u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2 + u_4 \mathbf{n}_4), \quad (\text{B.13})$$

where V_1 and V_3 denote the volumes of the triangles, $\{1, 2, 4\}$ and $\{3, 4, 2\}$, respectively. For the dual volume, we have the following geometric identities:

$$\mathbf{n}_1 = 2(\mathbf{n}_{12} + \mathbf{n}_{14}) + 4(\mathbf{n}_L + \mathbf{n}_R), \quad (\text{B.14})$$

$$\mathbf{n}_2 = 2(\mathbf{n}_R + \mathbf{n}_{12} + \mathbf{n}_{14}), \quad (\text{B.15})$$

$$\mathbf{n}_4 = 2(\mathbf{n}_L + \mathbf{n}_{12} + \mathbf{n}_{14}), \quad (\text{B.16})$$

where the vectors, \mathbf{n}_{12} , \mathbf{n}_{14} , \mathbf{n}_R , \mathbf{n}_L , are scaled normals of the dual volume as indicated in Figure B.2. Expanding the formula (B.13) using these identities, we obtain

$$\begin{aligned} \int_{dual} \text{grad } u \, dV &= \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \frac{u_1 + u_4}{2} \mathbf{n}_{14} + u_1 \mathbf{n}_L + u_1 \mathbf{n}_R \\ &+ \frac{1}{2} (\mathbf{n}_L + \mathbf{n}_{12}) u_4 + \frac{1}{2} (\mathbf{n}_R + \mathbf{n}_{14}) u_2 \\ &+ \frac{1}{16} \left(\frac{V_3}{V_1} - 1 \right) (u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2 + u_4 \mathbf{n}_4). \end{aligned} \quad (\text{B.17})$$

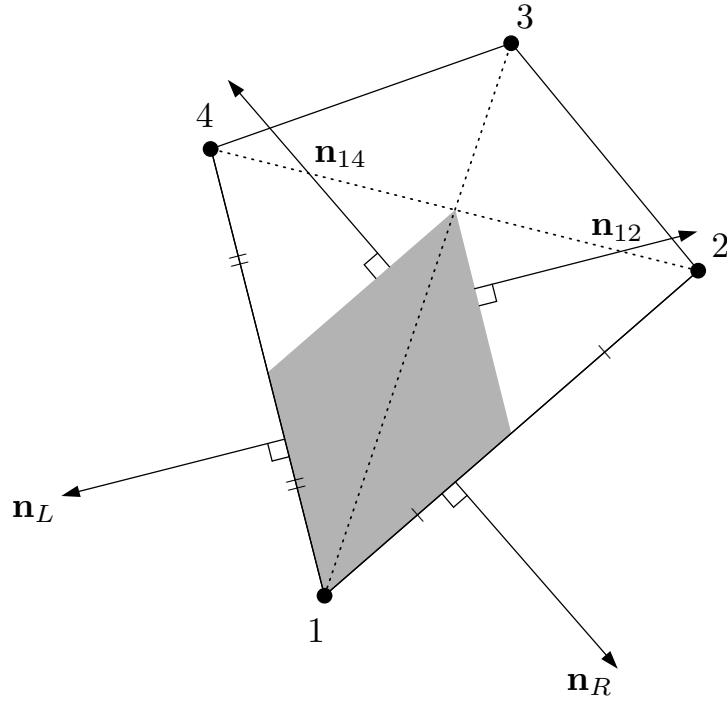


Figure B.3. Quadrilateral cell, a alternative dual-volume around a node 1, and scaled outward normals.

The terms on the first line is a common edge-based integration formula for a quadrilateral cell. The rest of the terms account for a deviation from a parallelogram; they vanish if the cell is a parallelogram. Consequently, the formula,

$$\int_{dual} \text{grad } u \, dV = \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \frac{u_1 + u_4}{2} \mathbf{n}_{14} + u_1 \mathbf{n}_L + u_1 \mathbf{n}_R, \quad (\text{B.18})$$

is exact for linear functions if the cell is a parallelogram.

B.1.3. Quadrilateral - Alternative Dual

If the dual volume is defined with the point of intersection of the diagonals instead of the centroid as in Figure B.3, a simpler formula can be obtained. Note that whereas the centroid is given simply by the average of the vertex coordinates, the intersection of the diagonals is given by

$$\frac{V_3}{V} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \frac{V_4}{V} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} + \frac{V_1}{V} \begin{bmatrix} x_3 \\ y_3 \end{bmatrix} + \frac{V_2}{V} \begin{bmatrix} x_4 \\ y_4 \end{bmatrix}, \quad (\text{B.19})$$

where V is the volume of the quadrilateral cell, and V_i is the volume of the triangle defined by i and two edge-connected vertices, e.g., V_1 is the volume of the triangle $\{1, 2, 4\}$. It is easy to show that the volume of the dual shown in Figure B.3 is precisely $\frac{1}{2}V_1$. Then, we multiply the Green-Gauss gradient over the triangle $\{1, 2, 4\}$ by the dual volume to get

$$\int_{dual} \text{grad } u \, dV = \frac{1}{4} (u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2 + u_4 \mathbf{n}_4), \quad (\text{B.20})$$

which is exact for linear functions. For the alternative dual-volume, we have the same identities as before,

$$\mathbf{n}_1 = 2(\mathbf{n}_{12} + \mathbf{n}_{14}) + 4(\mathbf{n}_L + \mathbf{n}_R), \quad (\text{B.21})$$

$$\mathbf{n}_2 = 2(\mathbf{n}_R + \mathbf{n}_{12} + \mathbf{n}_{14}), \quad (\text{B.22})$$

$$\mathbf{n}_4 = 2(\mathbf{n}_L + \mathbf{n}_{12} + \mathbf{n}_{14}). \quad (\text{B.23})$$

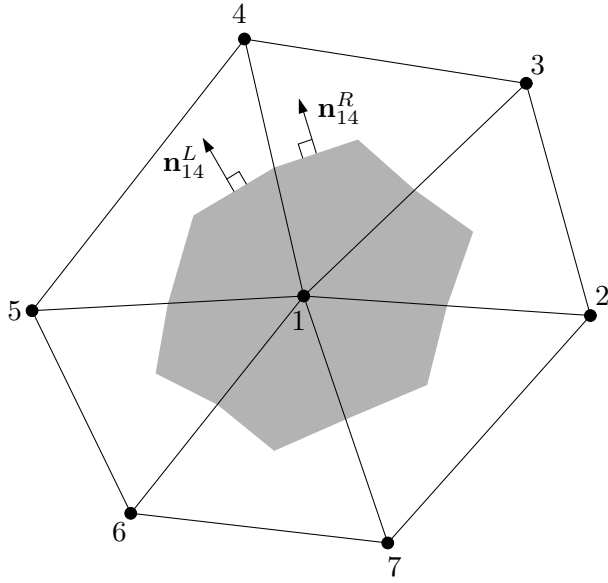


Figure B.4. Triangular grid, and scaled outward normals (not drawn to scale).

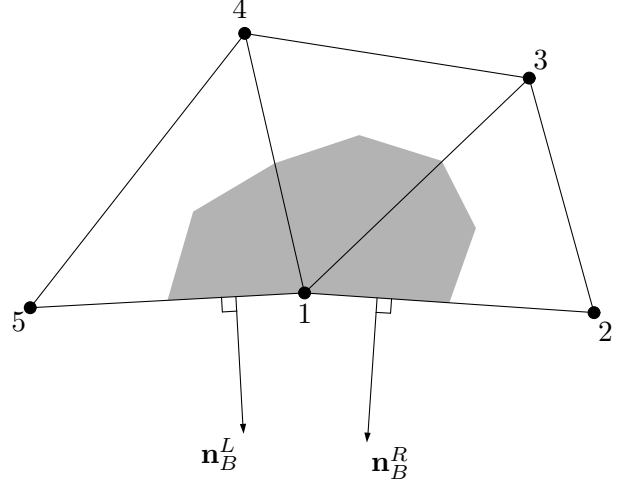


Figure B.5. Triangular grid on a boundary, and scaled outward normals (not drawn to scale).

Note that these relations are formally identical to those for the median dual volume, but the vectors, \mathbf{n}_{12} and \mathbf{n}_{14} , are different. Substituting these identities into the right hand side of Equation (B.20), we obtain

$$\begin{aligned} \int_{dual} \text{grad } u \, dV &= \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \frac{u_1 + u_4}{2} \mathbf{n}_{14} + u_1 \mathbf{n}_L + u_1 \mathbf{n}_R \\ &+ \frac{1}{2} (\mathbf{n}_L + \mathbf{n}_{12}) u_4 + \frac{1}{2} (\mathbf{n}_R + \mathbf{n}_{14}) u_2. \end{aligned} \quad (\text{B.24})$$

As before, the last two terms will vanish for a parallelogram.

B.1.4. Triangular Grids

For a triangular grid shown in Figure B.4, we can apply the formula (B.12) to each triangle to construct a finite-volume integral over the dual control volume around the node 1. Then, it is easy to see that all boundary contributions cancel out, and we obtain

$$\int_{dual} \text{grad } u \, dV = \sum_{k=2}^7 \frac{u_1 + u_k}{2} (\mathbf{n}_{1k}^L + \mathbf{n}_{1k}^R). \quad (\text{B.25})$$

This edge-based formula is, therefore, exact for any linear function, u .

If the node 1 is on a boundary as in Figure B.5, then we obtain, simply by collecting the contribution from each triangle,

$$\begin{aligned} \int_{dual} \text{grad } u \, dV &= \frac{u_1 + u_2}{2} \mathbf{n}_{12}^L + \frac{u_1 + u_3}{2} (\mathbf{n}_{13}^L + \mathbf{n}_{13}^R) + \frac{u_1 + u_4}{2} (\mathbf{n}_{14}^L + \mathbf{n}_{14}^R) \\ &+ \frac{u_1 + u_5}{2} \mathbf{n}_{15}^R + \left(\frac{5}{6} u_1 + \frac{1}{6} u_2 \right) \mathbf{n}_B^R + \left(\frac{5}{6} u_1 + \frac{1}{6} u_5 \right) \mathbf{n}_B^L, \end{aligned} \quad (\text{B.26})$$

which is exact for linear functions since it is a sum of the exact formula within each cell. The last two terms provide a boundary closure to make the edge-based integration formula exact for linear functions through the domain boundary [53, 20].

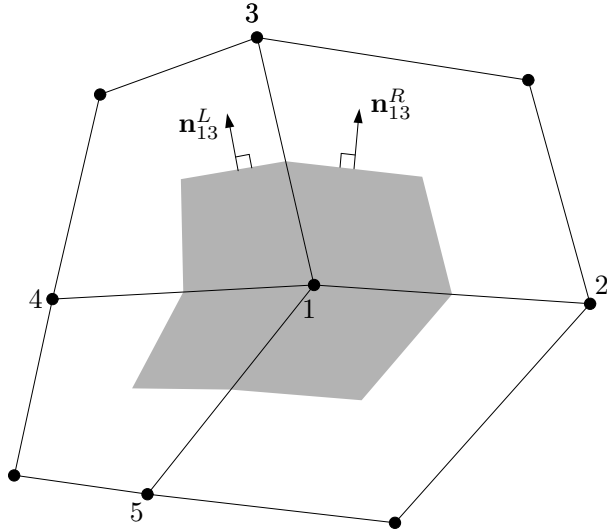


Figure B.6. Quadrilateral grid and scaled outward normals (not drawn to scale). Dual volume may be either the medial dual or the alternative dual (not drawn precisely).

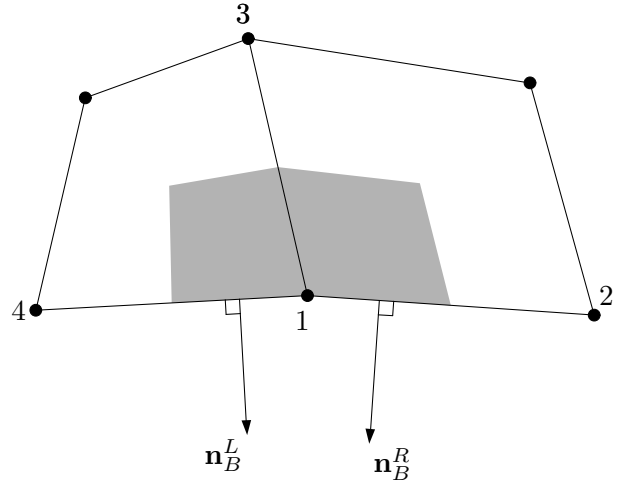


Figure B.7. Quadrilateral grid on a boundary and scaled outward normals (not drawn to scale). Dual volume may be either the medial dual or the alternative dual (not drawn precisely).

B.1.5. Quadrilateral Grids - Median Dual

For a general quadrilateral grid with median dual volumes as shown in Figure B.6, we may apply the formula (B.17) to each dual volume around the node 1 to obtain an integration formula that is exact for linear functions. The result is quite complicated but can be simplified if the quadrilaterals are all parallelograms. That is, we can apply the formula (B.18) to each dual volume to get

$$\int_{dual} \text{grad } u \, dV = \sum_{k=2}^5 \frac{u_1 + u_k}{2} (\mathbf{n}_{1k}^L + \mathbf{n}_{1k}^R), \quad (\text{B.27})$$

which is in the form of a common edge-based formula. This formula is exact for linear functions if the quadrilaterals around the node 1 are all parallelograms, or if those extra terms in the single-cell formula (B.17) happen to vanish when summed over the surrounding quadrilaterals. It should be noted that the exactness of the single-cell formula such as Equation (B.18) is not a necessary condition but only a sufficient condition for the exactness of the formula assembled around a node.

Next, consider a pair of quadrilaterals on a domain boundary as shown in Figure B.7. Applying the general formula (B.17) to each dual volume around the node 1, we obtain

$$\begin{aligned} \int_{dual} \text{grad } u \, dV &= \frac{u_1 + u_3}{2} (\mathbf{n}_{13}^L + \mathbf{n}_{13}^R) + \frac{u_1 + u_2}{2} \mathbf{n}_{12}^L + \frac{u_1 + u_4}{2} \mathbf{n}_{14}^R + u_1 \mathbf{n}_B^L + u_1 \mathbf{n}_B^R \\ &+ \frac{1}{2} (\mathbf{n}_B^R + \mathbf{n}_{13}^R) u_2 + \frac{1}{2} (\mathbf{n}_{12}^L + \mathbf{n}_{14}^R) u_3 + \frac{1}{2} (\mathbf{n}_B^L + \mathbf{n}_{13}^L) u_4 \\ &+ \frac{1}{16} \left(\frac{V_3^R}{V_1^R} - 1 \right) (u_1 \mathbf{n}_1^R + u_2 \mathbf{n}_2^R + u_3 \mathbf{n}_3^R) \\ &+ \frac{1}{16} \left(\frac{V_3^L}{V_1^L} - 1 \right) (u_1 \mathbf{n}_1^L + u_3 \mathbf{n}_3^L + u_4 \mathbf{n}_4^L), \end{aligned} \quad (\text{B.28})$$

where the superscripts, L and R , have been introduced to indicate the left and right quadrilaterals. If the quadrilaterals are parallelograms, this formula reduces to

$$\int_{dual} \text{grad } u \, dV = \frac{u_1 + u_3}{2} (\mathbf{n}_{13}^L + \mathbf{n}_{13}^R) + \frac{u_1 + u_2}{2} \mathbf{n}_{12}^L + \frac{u_1 + u_4}{2} \mathbf{n}_{14}^R + u_1 \mathbf{n}_B^L + u_1 \mathbf{n}_B^R. \quad (\text{B.29})$$

The last two terms provide a boundary closure to make the edge-based integration formula exact for linear functions (if the cells are all parallelograms) through the domain boundary [20]. Note that the use of parallelograms for cells adjacent to a domain boundary significantly simplifies the boundary closure. In addition to

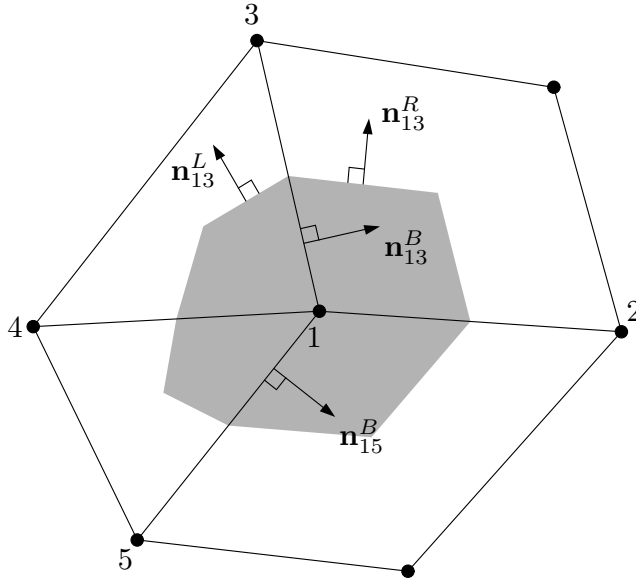


Figure B.8. General mixed-grid.

the simplicity of the formula, it has a great advantage that the boundary closure can be implemented without reference to interior nodes.

B.1.6. Quadrilateral Grids - Alternative Dual

If we employ the alternative dual volume instead of the median dual volume, we obtain, for *arbitrary* quadrilateral grids such as in Figure B.6), a common edge-based formula:

$$\int_{dual} \text{grad } u \, dV = \sum_{k=2}^5 \frac{u_1 + u_k}{2} (\mathbf{n}_{1k}^L + \mathbf{n}_{1k}^R). \quad (\text{B.30})$$

This formula is exact for linear functions for any quadrilateral grid. To prove it, first construct a formula for a pair of adjacent quadrilateral cells such as the one shown in Figure B.7:

$$\begin{aligned} \int_{dual\text{-pair}} \text{grad } u \, dV &= \frac{u_1 + u_3}{2} (\mathbf{n}_{13}^L + \mathbf{n}_{13}^R) + \frac{u_1 + u_2}{2} \mathbf{n}_{12}^L + \frac{u_1 + u_4}{2} \mathbf{n}_{14}^R + u_1 \mathbf{n}_B^L + u_1 \mathbf{n}_B^R \\ &+ \frac{1}{2} (\mathbf{n}_B^R + \mathbf{n}_{13}^R) u_2 + \frac{1}{2} (\mathbf{n}_{12}^L + \mathbf{n}_{14}^R) u_3 + \frac{1}{2} (\mathbf{n}_B^L + \mathbf{n}_{13}^L) u_4. \end{aligned} \quad (\text{B.31})$$

Then, apply this formula to all possible combinations of adjacent quadrilaterals in the stencil to find all extra terms sum up to zero, and finally divide the result by two to obtain Equation (B.30).

Equation (B.31) is, in effect, the formula on the boundary (see Figure B.7); it is significantly simpler than that for the median dual (B.28). Again, if the quadrilaterals are parallelograms, this formula reduces to

$$\int_{dual} \text{grad } u \, dV = \frac{u_1 + u_3}{2} (\mathbf{n}_{13}^L + \mathbf{n}_{13}^R) + \frac{u_1 + u_2}{2} \mathbf{n}_{12}^L + \frac{u_1 + u_4}{2} \mathbf{n}_{14}^R + u_1 \mathbf{n}_B^L + u_1 \mathbf{n}_B^R. \quad (\text{B.32})$$

B.1.7. Mixed Triangular/Quadrilateral Grids

Consider a mixed stencil shown in Figure B.8. We assume that the quadrilateral is a parallelogram. Applying the single-cell formulas (B.26) and (B.18) to triangles and quadrilaterals, respectively, we obtain

$$\int_{dual} \text{grad } u \, dV = \sum_{k=2}^5 \frac{u_1 + u_k}{2} (\mathbf{n}_{1k}^L + \mathbf{n}_{1k}^R) + \frac{1}{6} [(u_3 - u_1) \mathbf{n}_{13}^B + (u_5 - u_1) \mathbf{n}_{15}^B], \quad (\text{B.33})$$

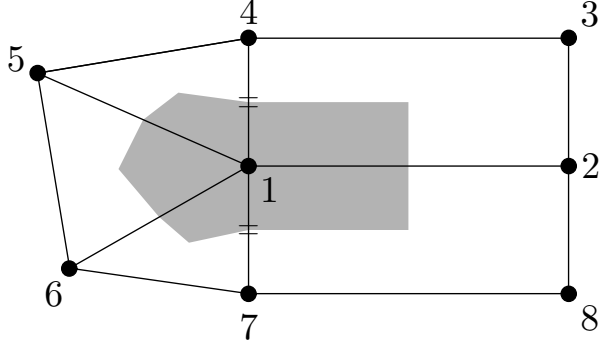


Figure B.9. Example of mixed grids for which the edge-based formula is exact for linear functions. The left half is composed of arbitrary triangular cells whereas the right half is composed of rectangular cells.

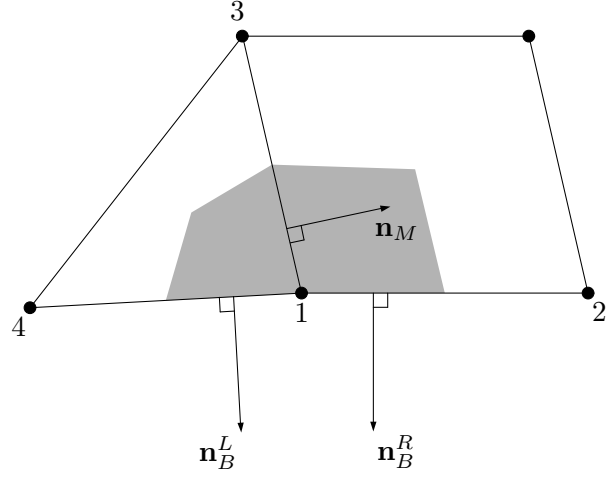


Figure B.10. Mixed grid on a boundary and scaled outward normals (not drawn to scale). The dual volume for the quadrilateral cell may be either the medial dual or the alternative dual.

where \mathbf{n}_{13}^B and \mathbf{n}_{15}^B denote the scaled normal to the edges $\{1, 3\}$ and $\{1, 5\}$, respectively. This formula is exact for linear functions if the quadrilaterals are parallelograms.

It is easy to see that the boundary term cannot vanish unless the edges $\{1, 3\}$ and $\{1, 5\}$ are aligned straight. Assume now that they are aligned (i.e., $\mathbf{n}_{13}^B \parallel \mathbf{n}_{15}^B$) and that the function is linear so that

$$u_3 - u_1 = \overline{\nabla} u \cdot \Delta \mathbf{l}_{13}, \quad (\text{B.34})$$

$$u_5 - u_1 = \overline{\nabla} u \cdot \Delta \mathbf{l}_{15}, \quad (\text{B.35})$$

where $\overline{\nabla} u$ is a non-zero constant gradient of the linear function, $\Delta \mathbf{l}_{13} = (x_3 - x_1, y_3 - y_1)$, and $\Delta \mathbf{l}_{15} = (x_5 - x_1, y_5 - y_1)$, then the boundary term can be written as

$$\frac{1}{6} [(u_3 - u_1)\mathbf{n}_{13}^B + (u_5 - u_1)\mathbf{n}_{15}^B] = \frac{1}{6} \overline{\nabla} u \cdot \left[\frac{\Delta \mathbf{l}_{13}}{\Delta l_{13}} \frac{\mathbf{n}_{13}^B}{\Delta l_{13}} \Delta l_{13}^2 + \frac{\Delta \mathbf{l}_{15}}{\Delta l_{15}} \frac{\mathbf{n}_{15}^B}{\Delta l_{15}} \Delta l_{15}^2 \right] \quad (\text{B.36})$$

$$= \frac{1}{6} \overline{\nabla} u \cdot \frac{\Delta \mathbf{l}_{13}}{\Delta l_{13}} [\Delta l_{13}^2 - \Delta l_{15}^2] \frac{\mathbf{n}_{13}^B}{\Delta l_{13}}, \quad (\text{B.37})$$

where $\Delta l_{13} = |\Delta \mathbf{l}_{13}|$ and $\Delta l_{15} = |\Delta \mathbf{l}_{15}|$, which thus vanishes only if

$$\Delta l_{13} = \Delta l_{15}. \quad (\text{B.38})$$

Therefore, the boundary term vanishes if the interface between a group of triangles and a group of parallelograms is a straight line and the node 1 is located at the midpoint of the interface. The stencil shown in Figure B.9 is such an example. If the node 1 is not located at the midpoint, the formula is not exact for linear functions, but can provide second-order accuracy for a smooth grid such that $\Delta l_{13} - \Delta l_{15} = O(h^2)$ as shown in Ref. [20].

The above argument is valid only for interior stencils. It does not apply to a stencil around a boundary node. Consider a stencil shown in Figure B.10. The cell on the right is assumed to be a parallelogram. Applying the single-cell formulas (B.26) and (B.18) to the triangle and the quadrilateral, respectively, we obtain

$$\begin{aligned} \int_{dual} \text{grad } u \, dV &= \frac{u_1 + u_2}{2} \mathbf{n}_{12}^L + \frac{u_1 + u_3}{2} (\mathbf{n}_{13}^L + \mathbf{n}_{13}^R) + \frac{u_1 + u_4}{2} \mathbf{n}_{14}^R \\ &+ \left(\frac{5}{6} u_1 + \frac{1}{6} u_4 \right) \mathbf{n}_B^L + u_1 \mathbf{n}_B^R + \frac{1}{6} (u_3 - u_1) \mathbf{n}_M. \end{aligned} \quad (\text{B.39})$$

The last term cannot vanish unless $u_1 = u_2$. If this contribution is ignored, the integration cannot be exact for linear functions.

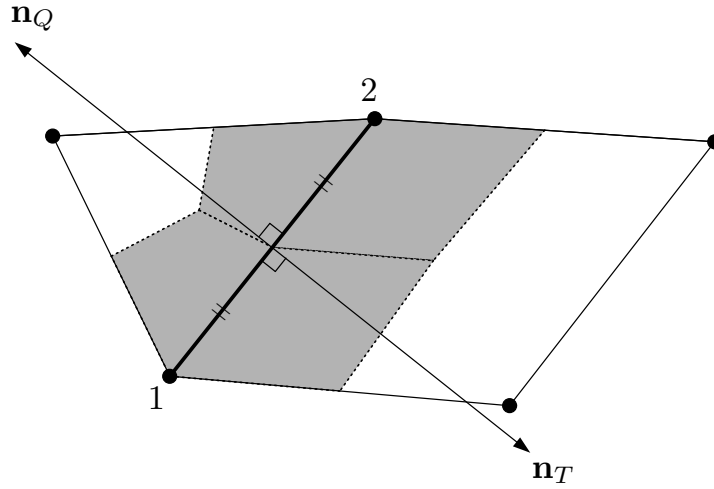


Figure B.11. Interface edge and scaled normals. $\mathbf{n}_T = -\mathbf{n}_Q$ and $|\mathbf{n}_T| = |\mathbf{n}_Q| = \Delta l_{12}$.

B.1.8. Linearity-Preserving Edge-Based Integration on General Mixed-Grids

One way to ensure the linearity-preserving property on general mixed-grids is to loop over the cells and apply a single-cell formula in each cell to accumulate contributions to the nodes. This strategy can be implemented efficiently as a loop over edges followed by a loop over boundary edges and interface-edges (those shared by a triangle and a quadrilateral). For nodes surrounded by triangles only or quadrilaterals only, the boundary contributions in the single-cell formula cancel out (if the quadrilaterals are parallelograms or the alternative dual volume is used). Therefore, the edge-based formulas (B.25) or (B.27) can be applied to perform the integration efficiently for these nodes. Then, the boundary contributions remain only across the interface-edges and the domain boundary. These contributions can be easily added in a loop over the interface-edges and the boundary edges. For example, for an interface-edge shown in Figure B.11, add the following to the node 1,

$$\frac{5}{6}u_1 \left(\frac{\mathbf{n}_T}{2} \right) + \frac{1}{6}u_2 \left(\frac{\mathbf{n}_T}{2} \right) + u_1 \left(\frac{\mathbf{n}_Q}{2} \right), \quad (\text{B.40})$$

and the following to the node 2,

$$\frac{5}{6}u_2 \left(\frac{\mathbf{n}_T}{2} \right) + \frac{1}{6}u_1 \left(\frac{\mathbf{n}_T}{2} \right) + u_2 \left(\frac{\mathbf{n}_Q}{2} \right). \quad (\text{B.41})$$

These extra contributions may be considered as a source term to each node, but they will vanish globally when summed over all nodes in the domain. Cancellation occurs along each interface-edge: the sum of the contributions (B.40) and (B.41) are zero. For integration of conservation laws, this means that a global conservation is guaranteed while exactness for linear fluxes is guaranteed locally around each node.

B.2. Three Dimensions

B.2.1. Tetrahedron

Consider a tetrahedral cell in Figure B.12. The dual volume around the node 1, which is indicated by the shaded area, is defined by the centroid of the cell, the centroids of the adjacent triangular faces, and the midpoints of the incident edges. The volume of the dual thus defined, V_{dual} , is then precisely $\frac{1}{4}$ of the volume of the cell, V :

$$V_{dual} = \frac{1}{4}V. \quad (\text{B.42})$$

If u is a linear function, the gradient is a global constant, which can be exactly expressed by the Green-Gauss gradient:

$$(\text{grad } u) = \frac{1}{3V} \sum_{i=1}^4 u_i \mathbf{n}_i, \quad (\text{B.43})$$

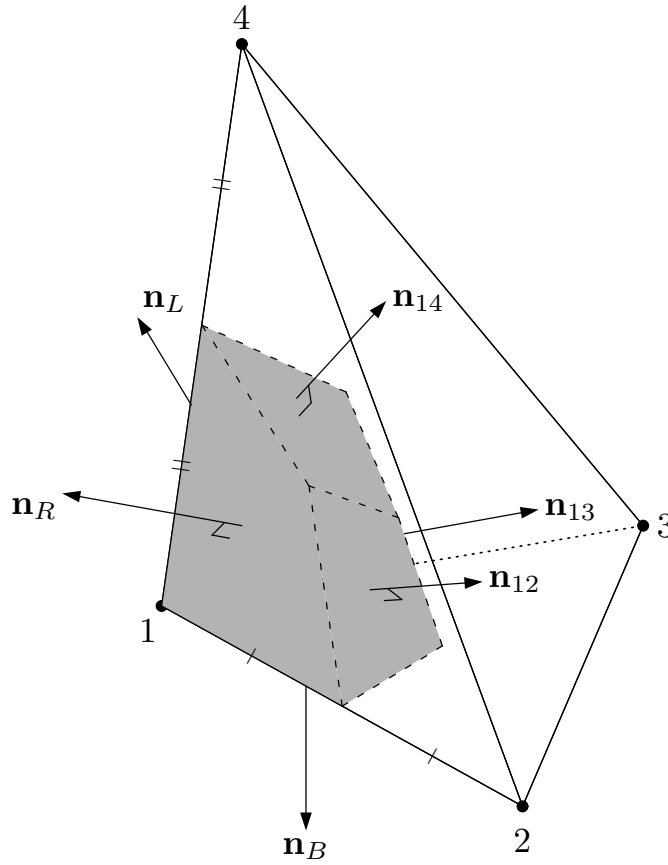


Figure B.12. Tetrahedral cell, a dual volume around the node 1, and scaled outward normals (not drawn to scale).

where u_i is the function value at the node i , and \mathbf{n}_i is the scaled inward normal of the triangular face opposite to the node i . Then, the integral (B.1) can be evaluated as follows:

$$\int_{dual} \text{grad } u \, dV = (\text{grad } u) \times \frac{1}{4}V = \frac{1}{12} (u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2 + u_3 \mathbf{n}_3 + u_4 \mathbf{n}_4). \quad (\text{B.44})$$

Substituting into Equation (B.44) the following geometric identities,

$$\mathbf{n}_1 = 6(\mathbf{n}_{12} + \mathbf{n}_{13} + \mathbf{n}_{14}) + 9(\mathbf{n}_L + \mathbf{n}_R + \mathbf{n}_B), \quad (\text{B.45})$$

$$\mathbf{n}_2 = 6\mathbf{n}_{12} + \frac{3}{2}(\mathbf{n}_R + \mathbf{n}_B), \quad (\text{B.46})$$

$$\mathbf{n}_3 = 6\mathbf{n}_{13} + \frac{3}{2}(\mathbf{n}_B + \mathbf{n}_L), \quad (\text{B.47})$$

$$\mathbf{n}_4 = 6\mathbf{n}_{14} + \frac{3}{2}(\mathbf{n}_L + \mathbf{n}_R), \quad (\text{B.48})$$

where \mathbf{n}_{12} , \mathbf{n}_{13} , \mathbf{n}_{14} , \mathbf{n}_L , \mathbf{n}_R , \mathbf{n}_B are scaled outward normals of the dual volume as indicated in Figure B.12, we arrive at the following finite-volume formula:

$$\begin{aligned} \int_{dual} \text{grad } u \, dV &= \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \frac{u_1 + u_3}{2} \mathbf{n}_{13} + \frac{u_1 + u_4}{2} \mathbf{n}_{14} \\ &+ \left(\frac{6}{8}u_1 + \frac{1}{8}u_3 + \frac{1}{8}u_4 \right) \mathbf{n}_L + \left(\frac{6}{8}u_1 + \frac{1}{8}u_4 + \frac{1}{8}u_2 \right) \mathbf{n}_R + \left(\frac{6}{8}u_1 + \frac{1}{8}u_2 + \frac{1}{8}u_3 \right) \mathbf{n}_B. \end{aligned} \quad (\text{B.49})$$

This formula is, therefore, exact for any linear function, u . Note that the boundary contributions are all evaluated with the same set of weights, $6/8$, $1/8$, and $1/8$, without reference to the fourth node. Similarly to the case of triangles, these weights have long been known [53], but this simple derivation does not seem available in the literature.

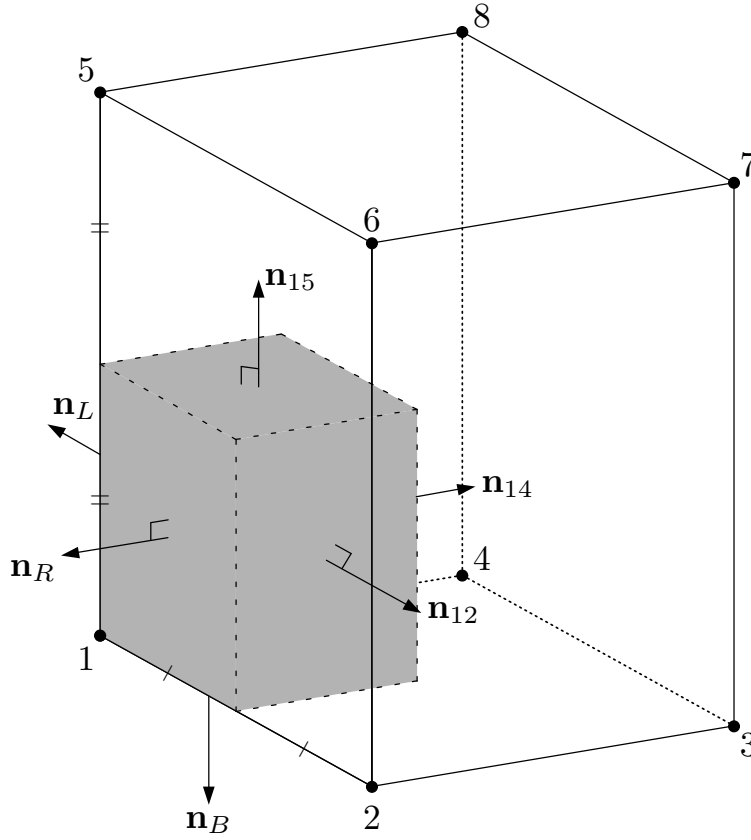


Figure B.13. Hexahedral (rectangular) cell, a dual volume around the node 1, and scaled outward normals (not drawn to scale).

B.2.2. Hexahedron

Consider a hexahedral cell in Figure B.13. Here, we assume no irregular perturbations and consider only rectangular cells; but the aspect ratio is arbitrary. The volume of the dual (the shaded volume), V_{dual} , is $\frac{1}{8}$ of the volume of the cell, V :

$$V_{dual} = \frac{1}{8}V. \quad (\text{B.50})$$

If u is a linear function, the gradient is a global constant, which can be expressed exactly by the Green-Gauss gradient over the tetrahedron, T , defined by the set of nodes, $\{i_T\} = \{1, 2, 4, 5\}$:

$$(\text{grad } u) = \frac{1}{3V_T} \sum_{i \in \{i_T\}} u_i \mathbf{n}_i, \quad (\text{B.51})$$

where $V_T = \frac{1}{6}V$ is the volume of the tetrahedron, u_i is the function value at the node i , and \mathbf{n}_i is the scaled inward normal of the triangular face opposite to the node i within the tetrahedron T . Then, the integral (B.1) can be evaluated as follows:

$$\int_{dual} \text{grad } u dV = (\text{grad } u) \times \frac{1}{8}V = \frac{1}{4}(u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2 + u_4 \mathbf{n}_4 + u_5 \mathbf{n}_5). \quad (\text{B.52})$$

Substituting into Equation (B.52) the following geometric identities,

$$\mathbf{n}_1 = 2(\mathbf{n}_{12} + \mathbf{n}_{14} + \mathbf{n}_{15}) + 4(\mathbf{n}_L + \mathbf{n}_R + \mathbf{n}_B), \quad \mathbf{n}_2 = 2\mathbf{n}_{12}, \quad \mathbf{n}_4 = 2\mathbf{n}_{14}, \quad \mathbf{n}_5 = 2\mathbf{n}_{15}, \quad (\text{B.53})$$

where \mathbf{n}_{12} , \mathbf{n}_{14} , \mathbf{n}_{15} , \mathbf{n}_L , \mathbf{n}_R , \mathbf{n}_B are scaled outward normals of the dual volume as indicated in Figure B.13, we arrive at the following finite-volume formula, which is exact for linear functions:

$$\int_{dual} \text{grad } u dV = \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \frac{u_1 + u_3}{2} \mathbf{n}_{13} + \frac{u_1 + u_4}{2} \mathbf{n}_{14} + u_1 \mathbf{n}_L + u_1 \mathbf{n}_R + u_1 \mathbf{n}_B. \quad (\text{B.54})$$

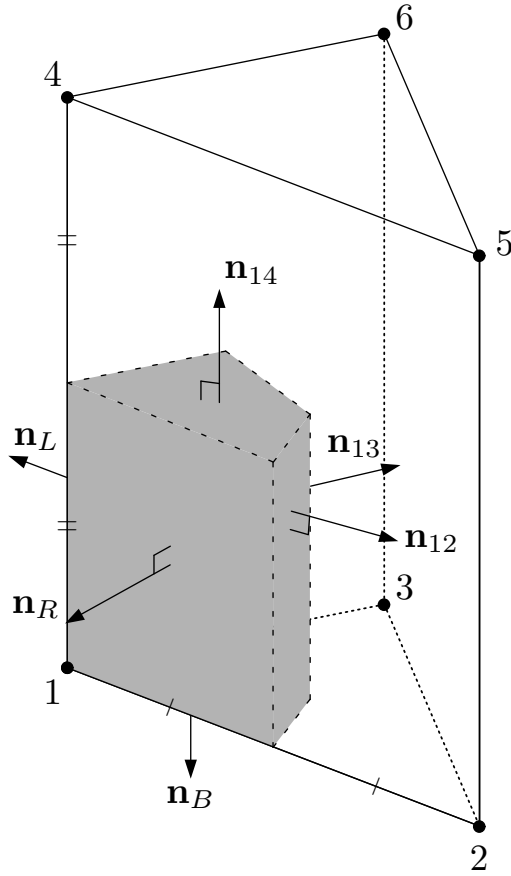


Figure B.14. Prismatic cell, a dual volume around the node 1, and scaled outward normals (not drawn to scale).

B.2.3. Prism

Consider a prismatic cell in Figure B.14. Here, we assume that the side faces are rectangles, and that the top and bottom triangular faces are congruent but of arbitrary shape. Under these assumptions, the volume of the dual (the shaded volume), V_{dual} , is exactly $\frac{1}{6}$ of the volume of the cell, V :

$$V_{dual} = \frac{1}{6}V. \quad (\text{B.55})$$

If u is a linear function, the gradient can be expressed exactly by the Green-Gauss gradient over the tetrahedron, T , defined by the set of nodes, $\{1, 2, 3, 4\}$:

$$(\text{grad } u) = \frac{1}{3V_T} \sum_{i=1}^4 u_i \mathbf{n}_i, \quad (\text{B.56})$$

where $V_T = \frac{1}{3}V$ is the volume of the tetrahedron, u_i is the function value at the node i , and \mathbf{n}_i is the scaled inward normal of the triangular face opposite to the node i within the tetrahedron T . Then, the integral (B.1) can be evaluated as follows:

$$\int_{dual} \text{grad } u \, dV = (\text{grad } u) \times \frac{1}{6}V = \frac{1}{6} (u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2 + u_3 \mathbf{n}_3 + u_4 \mathbf{n}_4). \quad (\text{B.57})$$

Substituting into Equation (B.57) the following geometric identities,

$$\mathbf{n}_1 = 3(\mathbf{n}_{12} + \mathbf{n}_{14} + \mathbf{n}_{15}) + 5(\mathbf{n}_L + \mathbf{n}_R) + 6\mathbf{n}_B, \quad (\text{B.58})$$

$$\mathbf{n}_2 = 3\mathbf{n}_{12} + \mathbf{n}_R, \quad (\text{B.59})$$

$$\mathbf{n}_3 = 3\mathbf{n}_{13} + \mathbf{n}_L, \quad (\text{B.60})$$

$$\mathbf{n}_4 = 3\mathbf{n}_{14}, \quad (\text{B.61})$$

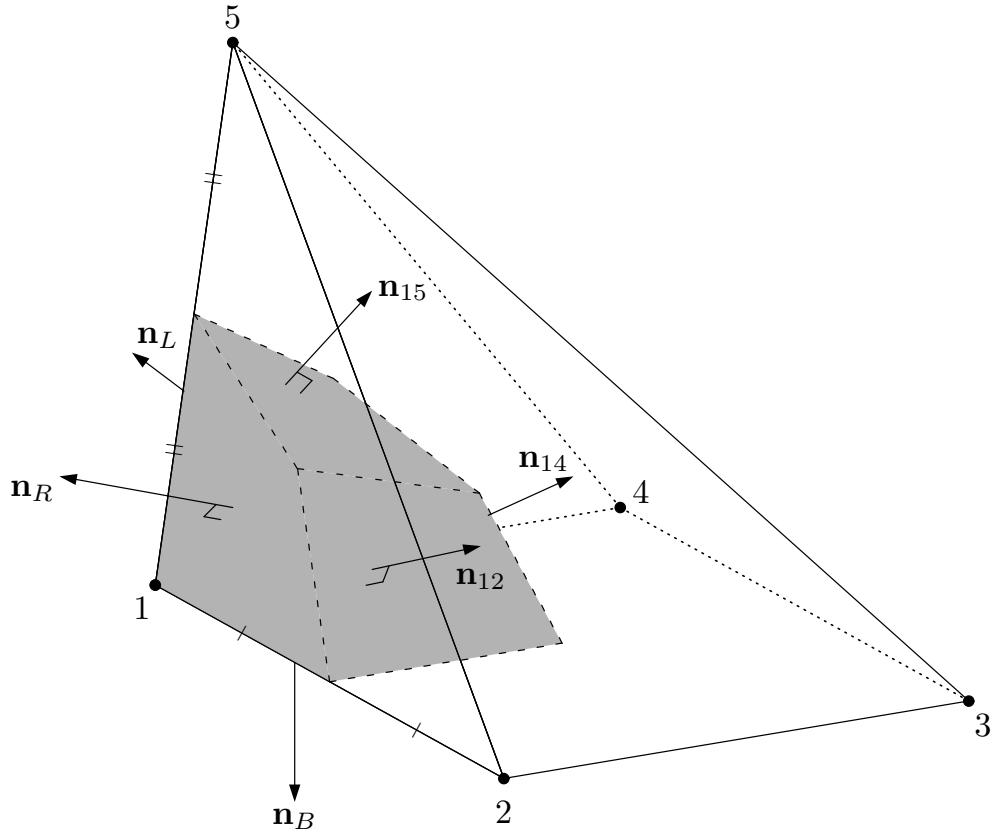


Figure B.15. Pyramidal cell, a dual volume around the node 1, and scaled outward normals (not drawn to scale). The centroid is defined as a point dividing the segment between the apex and the centroid of the bottom rectangle by 3 : 1, not the average of the nodal coordinates.

where \mathbf{n}_{12} , \mathbf{n}_{14} , \mathbf{n}_{15} , \mathbf{n}_L , \mathbf{n}_R , \mathbf{n}_B are scaled outward normals of the dual volume as indicated in Figure B.13, we arrive at the following finite-volume formula:

$$\int_{dual} \text{grad } u \, dV = \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \frac{u_1 + u_3}{2} \mathbf{n}_{13} + \frac{u_1 + u_4}{2} \mathbf{n}_{14} + \left(\frac{5}{6} u_1 + \frac{1}{6} u_2 \right) \mathbf{n}_R + \left(\frac{5}{6} u_1 + \frac{1}{6} u_3 \right) \mathbf{n}_L + u_1 \mathbf{n}_B. \quad (\text{B.62})$$

This formula is exact for any linear function, u . Compare the boundary contributions with those of triangular cell (B.12) and the hexahedral cell (B.54).

B.2.4. Pyramid

Consider a pyramidal cell in Figure B.14. Here, we only consider pyramids having a rectangular base. The dual control volume (the shaded volume) is defined by the centroid of the pyramid, the centroids of the adjacent triangular/rectangular faces, and the midpoints of the incident edges. Note that the centroid of the pyramid is not the average of the vertex coordinates, but the point that divides the segment between the apex and the centroid of the base in the ratio 3 : 1. For nodes on the base, $\{1, 2, 3, 4\}$, the volume of the dual, V_{dual} , is precisely $\frac{3}{16}$ of the volume of the cell, V :

$$V_{dual_i} = \frac{3}{16} V, \quad (\text{B.63})$$

where $dual_i$ indicates the dual volume around the node i . The apex node, 5, will be considered later (the dual volume is $\frac{1}{4}V$). Again, if u is a linear function, the gradient can be expressed exactly by the Green-Gauss gradient over a tetrahedron, T , defined by the set of nodes $\{i_T\} = \{1, 2, 4, 5\}$:

$$(\text{grad } u) = \frac{1}{3V_T} \sum_{i \in \{i_T\}} u_i \mathbf{n}_i, \quad (\text{B.64})$$

where $V_T = \frac{1}{2}V$ is the volume of the tetrahedron, u_i is the function value at node i , and \mathbf{n}_i is the scaled inward normal of the triangular face opposite to the node i within the tetrahedron T . Then, the dual integral (B.1) around the node 1 can be evaluated as follows:

$$\int_{dual_1} \text{grad } u \, dV = (\text{grad } u) \times \frac{3}{16}V = \frac{1}{8}(u_1 \mathbf{n}_1 + u_2 \mathbf{n}_2 + u_4 \mathbf{n}_4 + u_5 \mathbf{n}_5). \quad (\text{B.65})$$

Substituting into Equation (B.65) the following geometric identities,

$$\mathbf{n}_1 = 4(\mathbf{n}_{12} + \mathbf{n}_{14} + \mathbf{n}_{15}) + 7(\mathbf{n}_L + \mathbf{n}_R) + 6\mathbf{n}_B, \quad (\text{B.66})$$

$$\mathbf{n}_2 = 4\mathbf{n}_{12} + \mathbf{n}_B, \quad (\text{B.67})$$

$$\mathbf{n}_4 = 4\mathbf{n}_{14} + \mathbf{n}_B, \quad (\text{B.68})$$

$$\mathbf{n}_5 = 4\mathbf{n}_{15} + \mathbf{n}_L + \mathbf{n}_R, \quad (\text{B.69})$$

where \mathbf{n}_{12} , \mathbf{n}_{14} , \mathbf{n}_{15} , \mathbf{n}_L , \mathbf{n}_R , \mathbf{n}_B are scaled outward normals of the dual volume as indicated in Figure B.15, we arrive at the edge-based finite-volume formula:

$$\begin{aligned} \int_{dual_1} \text{grad } u \, dV &= \frac{u_1 + u_2}{2} \mathbf{n}_{12} + \frac{u_1 + u_4}{2} \mathbf{n}_{14} + \frac{u_1 + u_5}{2} \mathbf{n}_{15} \\ &+ \left(\frac{7}{8}u_1 + \frac{1}{8}u_5 \right) \mathbf{n}_L + \left(\frac{7}{8}u_1 + \frac{1}{8}u_5 \right) \mathbf{n}_R + \left(\frac{6}{8}u_1 + \frac{1}{8}u_2 + \frac{1}{8}u_4 \right) \mathbf{n}_B. \end{aligned} \quad (\text{B.70})$$

This formula is exact for any linear function, u . The last term is similar to the boundary term for a tetrahedral cell. See Equation (B.12).

The integration formula for the dual volume around the node 5 can be derived by subtracting the dual integrations around the nodes, 1, 2, 3, and 4, (applying the formula derived above) from the Green-Gauss gradient over the pyramidal cell. The result is

$$\begin{aligned} \int_{dual_5} \text{grad } u \, dV &= \frac{u_5 + u_1}{2} \mathbf{n}_{51} + \frac{u_5 + u_2}{2} \mathbf{n}_{52} + \frac{u_5 + u_3}{2} \mathbf{n}_{53} + \frac{u_5 + u_4}{2} \mathbf{n}_{54} \\ &+ \left(\frac{6}{8}u_5 + \frac{1}{8}u_1 + \frac{1}{8}u_2 \right) \mathbf{n}_a + \left(\frac{6}{8}u_5 + \frac{1}{8}u_2 + \frac{1}{8}u_3 \right) \mathbf{n}_b \\ &+ \left(\frac{6}{8}u_5 + \frac{1}{8}u_3 + \frac{1}{8}u_4 \right) \mathbf{n}_c + \left(\frac{6}{8}u_5 + \frac{1}{8}u_4 + \frac{1}{8}u_1 \right) \mathbf{n}_d, \end{aligned} \quad (\text{B.71})$$

where $\mathbf{n}_{51} = -\mathbf{n}_{15}$ is the scaled normal of the dual-face pointing from the node 5 to the node 1 (similarly for others), and \mathbf{n}_a is the scaled outward normal of the dual face on the triangular face $\{1, 2, 5\}$, \mathbf{n}_b on the face $\{2, 3, 5\}$, \mathbf{n}_c on the face $\{3, 4, 5\}$, and \mathbf{n}_d on the face $\{4, 1, 5\}$. This formula is exact for linear functions since all formulas used are exact for linear functions. The boundary coefficients are identical to those in the tetrahedral formula (B.12).

B.2.5. Hexahedral Grids and Tetrahedral Grids

Just like the triangular grid case, the edge-based formula assembled at a node (by applying the single-cell formula to each cell in the stencil) is exact for linear functions for hexahedral (rectangular) grids as well as for arbitrary tetrahedral grids without any boundary contributions since the boundary contributions will cancel out for all faces that shared by two adjacent cells. If the node is on a boundary, the edge-based formula is still exact for linear functions, as long as boundary-face contributions that do not cancel out are retained as in Equation (B.26) for triangular grids.

B.2.6. Mixed Grids in Three Dimensions

For mixed grids, the boundary contributions from single-cell formulas do not cancel in general at faces between two different types of cells. They will vanish if the interface, e.g., a face between prismatic/hexahedral, prismatic/pyramidal, hexahedral/pyramidal, prismatic/tetrahedral, forms a *planer and symmetric* two-dimensional stencil. To ensure the linearity-preserving property at all dual volumes, it is necessary to treat all the interfaces shared by different types of cells as boundary faces, and compute the boundary contribution from each cell, in a manner described for two-dimensional mixed-grids in Section B.1.7. Again, for integration of conservation laws, this procedure guarantees a global conservation while retaining exactness for linear fluxes around each node.

Appendix C. Average-Least-Squares Schemes

We derive the edge-normal and face-tangent average-least-squares schemes to demonstrate the special choices of α discussed in the context of edge-based schemes in Section 5.3.2. Taking a common approach, in the stencil shown in Figure 5.1, we consider constructing an interface gradient, $\nabla u|_{jk}$, and evaluating the diffusive flux directly as $\phi_{jk} = -\nu \nabla u|_{jk} \cdot \hat{\mathbf{n}}_{jk}$. We then separate the diffusive flux into the consistent and damping terms to find the corresponding α by comparing with the flux (5.29). We also show that the edge-normal/face-tangent average-least-squares scheme can be implemented without the edge-normal/face-tangent vector (in both two and three dimensions).

C.1. Edge-Normal Scheme

In the edge-normal gradient construction, we define the interface gradient as

$$\nabla u|_{jk} = (\partial^e u) \hat{\mathbf{e}}_{jk} + (\partial^\perp u) \hat{\mathbf{e}}_{jk}^\perp, \quad (\text{C.1})$$

where $\partial^e u = (u_k - u_j)/\Delta l_{jk}$ is the edge-derivative, $\partial^\perp u$ is an edge-normal derivative to be specified, and $\hat{\mathbf{e}}_{jk}^\perp$ denotes the unit vector perpendicular to the edge-vector $\hat{\mathbf{e}}_{jk}$, i.e., $\hat{\mathbf{e}}_{jk}^\perp \cdot \hat{\mathbf{e}}_{jk} = 0$. If we employ the averaged least-squares gradient,

$$\overline{\nabla u} = \frac{1}{2} [(\nabla u)_k + (\nabla u)_j], \quad (\text{C.2})$$

to define $\partial^\perp u$ as

$$\partial^\perp u = \overline{\nabla u} \cdot \hat{\mathbf{e}}_{jk}^\perp, \quad (\text{C.3})$$

then we obtain

$$\nabla u|_{jk} = (\partial^e u) \hat{\mathbf{e}}_{jk} + (\overline{\nabla u} \cdot \hat{\mathbf{e}}_{jk}^\perp) \hat{\mathbf{e}}_{jk}^\perp \quad (\text{C.4})$$

$$= (\partial^e u) \hat{\mathbf{e}}_{jk} + \overline{\nabla u} - (\overline{\nabla u} \cdot \hat{\mathbf{e}}_{jk}) \hat{\mathbf{e}}_{jk}, \quad (\text{C.5})$$

which leads to the following diffusive flux,

$$\phi_{jk} = -\nu \nabla u|_{jk} \cdot \hat{\mathbf{n}}_{jk} = -\nu [(\partial^e u) \hat{\mathbf{e}}_{jk} + \overline{\nabla u} - (\overline{\nabla u} \cdot \hat{\mathbf{e}}_{jk}) \hat{\mathbf{e}}_{jk}] \cdot \hat{\mathbf{n}}_{jk}. \quad (\text{C.6})$$

This flux defines the edge-normal average-least-squares scheme [11, 25, 54]. Note that the edge-normal vector has been eliminated and therefore the edge-normal average-least-squares scheme can be implemented without the edge-normal vector. It is easy to show that the same is true in three dimensions (add one more averaged least-squares gradient projected along another orthogonal edge-normal vector to $\nabla u|_{jk}$, and eliminate it as is done above in the step from Equation (C.4) to Equation (C.5)). Hence, Equation (C.6) is valid for both two- and three-dimensional grids. Finally, by rearranging Equation (C.6) as

$$\phi_{jk} = -\nu \overline{\nabla u} \cdot \hat{\mathbf{n}}_{jk} - (\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}) [\partial^e u - \overline{\nabla u} \cdot \hat{\mathbf{e}}_{jk}], \quad (\text{C.7})$$

we find that it is equivalent to taking

$$\alpha = (\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}) |\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}|, \quad (\text{C.8})$$

in the diffusive flux (5.29) derived in this paper.

C.2. Face-Tangent Scheme

In the face-tangent construction, we seek the interface gradient in the form,

$$\nabla u|_{jk} = a_n \hat{\mathbf{n}}_{jk} + (\partial^f u) \hat{\mathbf{f}}_{jk}, \quad (\text{C.9})$$

where a_n is a component to be determined, $\hat{\mathbf{f}}_{jk}$ denotes a unit vector along the face, and $\partial^f u$ is a derivative along the face (to be specified). The face-tangent direction is taken to be orthogonal to the directed-area vector:

$\hat{\mathbf{f}}_{jk} \cdot \hat{\mathbf{n}}_{jk} = 0$. The component, a_n , is determined by requiring the interface gradient to give the edge-derivative when projected along $\hat{\mathbf{e}}_{jk}$:

$$\nabla u|_{jk} \cdot \hat{\mathbf{e}}_{jk} = a_n (\hat{\mathbf{n}}_{jk} \cdot \hat{\mathbf{e}}_{jk}) + \partial^f u \left(\hat{\mathbf{f}}_{jk} \cdot \hat{\mathbf{e}}_{jk} \right) = \partial^e u, \quad (\text{C.10})$$

and thus

$$a_n = \frac{1}{\hat{\mathbf{n}}_{jk} \cdot \hat{\mathbf{e}}_{jk}} \left[\partial^e u - \left(\hat{\mathbf{f}}_{jk} \cdot \hat{\mathbf{e}}_{jk} \right) \partial^f u \right]. \quad (\text{C.11})$$

This completes the construction of the interface gradient. The diffusive flux is then computed as

$$\phi_{jk} = -\nu \nabla u|_{jk} \cdot \hat{\mathbf{n}}_{jk} = -\frac{\nu}{\hat{\mathbf{n}}_{jk} \cdot \hat{\mathbf{e}}_{jk}} \left[\partial^e u - \left(\hat{\mathbf{f}}_{jk} \cdot \hat{\mathbf{e}}_{jk} \right) \partial^f u \right] \cdot \hat{\mathbf{n}}_{jk}. \quad (\text{C.12})$$

If the face-derivative, $\partial^f u$, is evaluated by the difference between node-averaged solutions, the resulting scheme will be the so-called node-averaging scheme [11,83], while if it is evaluated by a least-squares gradient constructed directly at the face, the scheme will be the face-least-squares scheme [11]. Now, if we use the averaged least-squares gradient to define $\partial^f u$ as

$$\partial^f u = \overline{\nabla u} \cdot \hat{\mathbf{f}}_{jk}, \quad (\text{C.13})$$

then the diffusive flux is given by

$$\phi_{jk} = -\frac{\nu}{\hat{\mathbf{n}}_{jk} \cdot \hat{\mathbf{e}}_{jk}} \left[\partial^e u - \left(\hat{\mathbf{f}}_{jk} \cdot \hat{\mathbf{e}}_{jk} \right) \overline{\nabla u} \cdot \hat{\mathbf{f}}_{jk} \right] \cdot \hat{\mathbf{n}}_{jk}. \quad (\text{C.14})$$

This is the face-tangent average-least-squares scheme. The scheme can be rewritten, by the following geometric identity,

$$\left(\hat{\mathbf{f}}_{jk} \cdot \hat{\mathbf{e}}_{jk} \right) \hat{\mathbf{f}}_{jk} = \hat{\mathbf{e}}_{jk} - (\hat{\mathbf{n}}_{jk} \cdot \hat{\mathbf{e}}_{jk}) \hat{\mathbf{n}}_{jk}, \quad (\text{C.15})$$

in the form independent of the face-tangent direction [55]:

$$\phi_{jk} = -\nu \overline{\nabla u} \cdot \hat{\mathbf{n}}_{jk} - \frac{\nu}{\hat{\mathbf{n}}_{jk} \cdot \hat{\mathbf{e}}_{jk}} \left[\partial^e u - \overline{\nabla u} \cdot \hat{\mathbf{e}}_{jk} \right]. \quad (\text{C.16})$$

Hence, the face-tangent average-least-squares scheme can be implemented without any face-tangent vector. It can be shown that this is true in three dimensions also (add one more averaged least-squares gradient projected along another orthogonal face-tangent vector to $\nabla u|_{jk}$, and eliminate it by using the three-dimensional version of the identity (C.15)). Hence, Equation (C.16) is valid for both two- and three-dimensional grids. Finally, it follows from Equation (C.16) that the face-tangent average-least-squares scheme is equivalent to taking

$$\alpha = \frac{|\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}|}{\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}}, \quad (\text{C.17})$$

in the diffusive flux (5.29) derived in this paper.

Appendix D. Edge-Difference Form of the Braaten-Connell Diffusion scheme

Here, we derive an edge-difference form of the edge-based compact scheme with the Green-Gauss gradients described in Section 5.3.3 for a triangular grid: the Braaten-Connell diffusion scheme [57]. Consider the stencil shown in Figure D.1. We focus on the diffusive flux across the edge connecting the nodes, j and k . The Green-Gauss gradient, $\nabla \tilde{u}_{jk}$, defined over the quadrilateral $\{j, k_r, k, k_\ell\}$, can be written as

$$\nabla \tilde{u}_{jk} = \frac{1}{2(V^L + V^R)} \left[u_j (\mathbf{n}_j^L + \mathbf{n}_j^R) + u_k (\mathbf{n}_k^L + \mathbf{n}_k^R) + u_{k_\ell} \mathbf{n}_{k_\ell}^L + u_{k_r} \mathbf{n}_{k_r}^R \right], \quad (\text{D.1})$$

where V^E is the volume of the element $E \in \{L, R\}$, and \mathbf{n}_i^E is the scaled inward normal of the face opposite to a node $i \in \{j, k_r, k, k_\ell\}$ in the element $E \in \{L, R\}$. The diffusive flux across the edge is then computed as

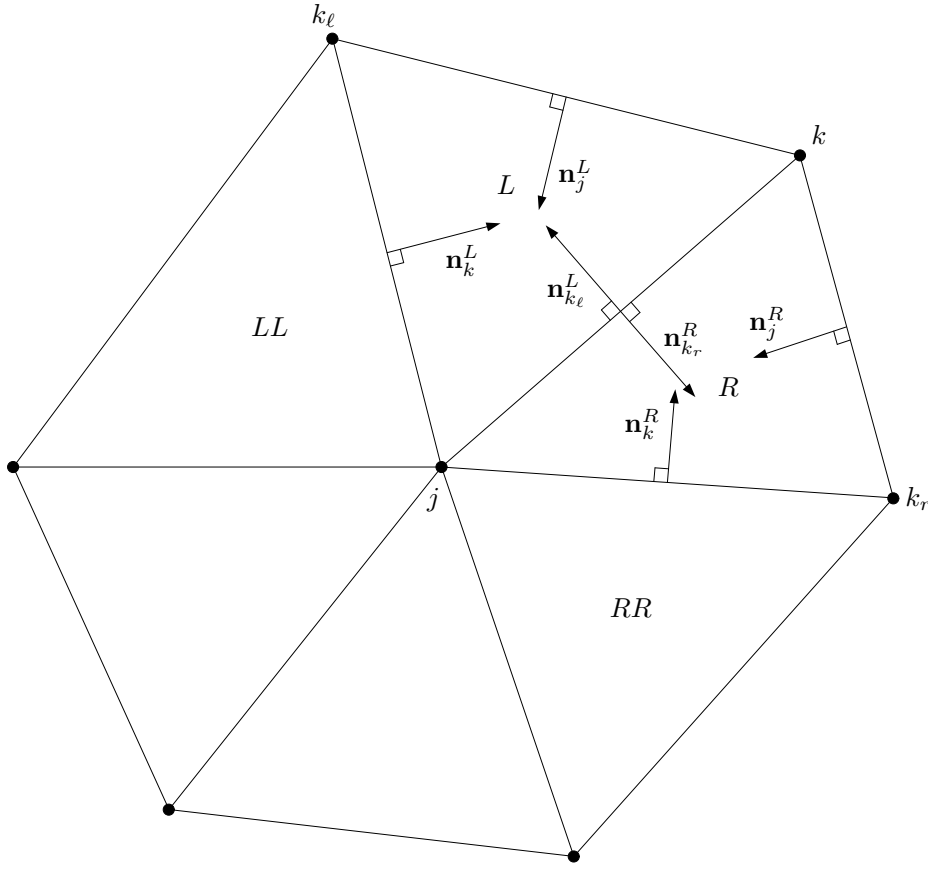


Figure D.1. Triangular stencil for the Braaten-Connell scheme. Normals are not drawn to scale.

$\nu \nabla \tilde{u}_{jk} \cdot \mathbf{n}_{jk}$, where \mathbf{n}_{jk} is the directed area vector across the edge. Collecting contributions from all neighbors, we obtain the following edge-based diffusion scheme:

$$\frac{du_j}{dt} = \frac{1}{V_j} \sum_{k \in \{k_j\}} \nu \nabla \tilde{u}_{jk} \cdot \mathbf{n}_{jk}, \quad (\text{D.2})$$

where V_j is the median dual volume around the node j , and $\{k_j\}$ denotes a set of neighbors. This scheme corresponds to the two-dimensional version of the Braaten-Connell scheme [57].

We now show that the Braaten-Connell scheme can be written in the edge-difference form:

$$\frac{du_j}{dt} = \frac{1}{V_j} \sum_{k \in \{k_j\}} C_{jk}(u_k - u_j). \quad (\text{D.3})$$

Making use of the following geometric identities,

$$\mathbf{n}_k^L + \mathbf{n}_k^R = -(\mathbf{n}_j^L + \mathbf{n}_j^R), \quad (\text{D.4})$$

$$\mathbf{n}_{k_\ell}^L + \mathbf{n}_{k_r}^R = \mathbf{0}, \quad (\text{D.5})$$

$$\mathbf{n}_{jk} = -\frac{1}{3}(\mathbf{n}_j^L + \mathbf{n}_j^R), \quad (\text{D.6})$$

we write the Green-Gauss gradient as

$$\nabla \tilde{u}_{jk} = \frac{1}{2(V^L + V^R)} [3\mathbf{n}_{jk}(u_k - u_j) + \mathbf{n}_{k_\ell}^L(u_{k_\ell} - u_j) + \mathbf{n}_{k_r}^R(u_{k_r} - u_j)]. \quad (\text{D.7})$$

We thus have

$$\nabla \tilde{u}_{jk} \cdot \mathbf{n}_{jk} = \frac{1}{2(V^L + V^R)} [3\mathbf{n}_{jk} \cdot \mathbf{n}_{jk}(u_k - u_j) + \mathbf{n}_{k_\ell}^L \cdot \mathbf{n}_{jk}(u_{k_\ell} - u_j) + \mathbf{n}_{k_r}^R \cdot \mathbf{n}_{jk}(u_{k_r} - u_j)]. \quad (\text{D.8})$$

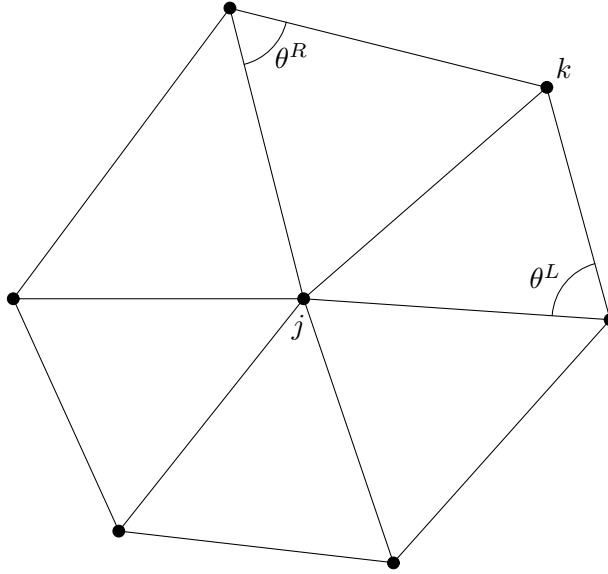


Figure E.1. Stencil and notations for the edge-based form of the Galerkin scheme (E.1).

Using this to rearrange the sum in Equation (D.2), we obtain

$$\frac{du_j}{dt} = \frac{1}{V_j} \sum_{k \in \{k_j\}} \frac{1}{2} \left[\frac{3 \mathbf{n}_{jk} \cdot \mathbf{n}_{jk}}{V^L + V^R} + \frac{\mathbf{n}_k^L \cdot \mathbf{n}_{jk_\ell}}{V^{LL} + V^L} + \frac{\mathbf{n}_k^R \cdot \mathbf{n}_{jk_r}}{V^R + V^{RR}} \right] (u_k - u_j), \quad (\text{D.9})$$

where LL/RR indicates the element adjacent to L/R sharing the node j but not R/L (see Figure D.1). Therefore, the Braaten-Connel scheme is in the form (D.3) with

$$C_{jk} = \frac{1}{2} \left[\frac{3 \mathbf{n}_{jk} \cdot \mathbf{n}_{jk}}{V^L + V^R} + \frac{\mathbf{n}_k^L \cdot \mathbf{n}_{jk_\ell}}{V^{LL} + V^L} + \frac{\mathbf{n}_k^R \cdot \mathbf{n}_{jk_r}}{V^R + V^{RR}} \right]. \quad (\text{D.10})$$

Note that because of the dependence on the elements LL and RR , we generally have

$$C_{jk} \neq C_{kj}. \quad (\text{D.11})$$

It is, therefore, more convenient to assemble the coefficients, C_{jk} , by making a loop over edges, and distributing all terms in the formula (D.8) to the node j , and negative of each term to the corresponding node in $\{k_r, k, k_\ell\}$. Note that the coefficient C_{jk} is not guaranteed to be positive on arbitrary grids. See Appendix E for numerical results.

Appendix E. Inconsistency of the Galerkin Scheme under Positivity Enforcement

It is well known that the Galerkin scheme can be written in the edge-based form (see Refs. [11, 75]):

$$\frac{du_j}{dt} = \frac{1}{2V_j} \sum_{k \in \{k_j\}} (\cot \theta^L + \cot \theta^R) (u_k - u_j), \quad (\text{E.1})$$

where V_j is the median dual volume around the node j , $\{k_j\}$ denotes a set of neighbors, and θ^L and θ^R are angles (left and right with respect to the node k) in the quadrilateral formed by the nodes: j , k , and their common neighbors (see Figure E.1). Suppose that a linear function, w , is substituted into the scheme (E.1). Then, we can write the Galerkin scheme as

$$\frac{dw_j}{dt} = \frac{1}{V_j} \sum_{k \in \{k_j\}} \mathbf{c}_{jk} \cdot \nabla w, \quad (\text{E.2})$$

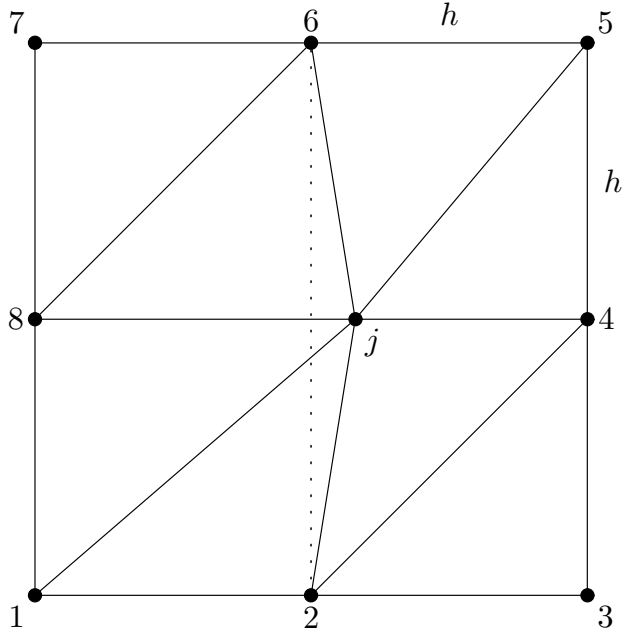


Figure E.2. Perturbed stencil.

where ∇w is a global constant, and

$$\mathbf{c}_{jk} = \frac{1}{2} (\cot \theta^L + \cot \theta^R) \Delta \mathbf{l}_{jk}, \quad \Delta \mathbf{l}_{jk} = (x_k - x_j, y_k - y_j). \quad (\text{E.3})$$

It can be shown that

$$\sum_{k \in \{k_j\}} \mathbf{c}_{jk} = \mathbf{0}, \quad (\text{E.4})$$

so that we have

$$\frac{dw_j}{dt} = 0, \quad (\text{E.5})$$

for linear functions. That is, the Galerkin scheme preserves arbitrary linear functions. This is consistent with the diffusion equation.

It follows from Equation (E.1) that the discretization is positive if $\theta^L + \theta^R < \pi$, or equivalently if the triangulation is Delaunay [75]. Many practical grids, however, do not satisfy this condition. Even for a regular triangulation as shown in Figure E.2, any nonzero perturbation to the node j will generate a negative coefficient, e.g., for edge $\{1, j\}$ in the figure. The Galerkin scheme then loses positivity, and often leads to convergence difficulties and/or numerical oscillations as shown in Section 6.2.2. A popular practical remedy is to enforce the positivity by ignoring any negative contribution:

$$\frac{du_j}{dt} = \frac{1}{2V_j} \sum_{k \in \{k_j\}} \max(0, \cot \theta^L + \cot \theta^R) (u_k - u_j). \quad (\text{E.6})$$

But this technique, although effective, leads to inconsistency. If the enforcement is activated, the scheme will lose the property (E.4). It means that the numerical solution will evolve by nonzero first-order derivatives, and thus the scheme is not consistent with the diffusion equation. We tested the Galerkin scheme with the positivity enforcement for the isotropic test case in Section 6.2.1 and the anisotropic test case in Section 6.2.2. For the anisotropic case, we added 8 additional fine grids: 145x145, 153x153, 161x161, 169x169, 177x177, 185x185, 193x193, 201x201 grids. Results are shown in Figure E.3. As expected, the errors do not converge with the grid refinement. In the anisotropic case, it appears second-order accurate for the first 14 grids, but the accuracy deteriorates thereafter. In the isotropic case, the positivity enforcement was activated on all grids for 17 % of

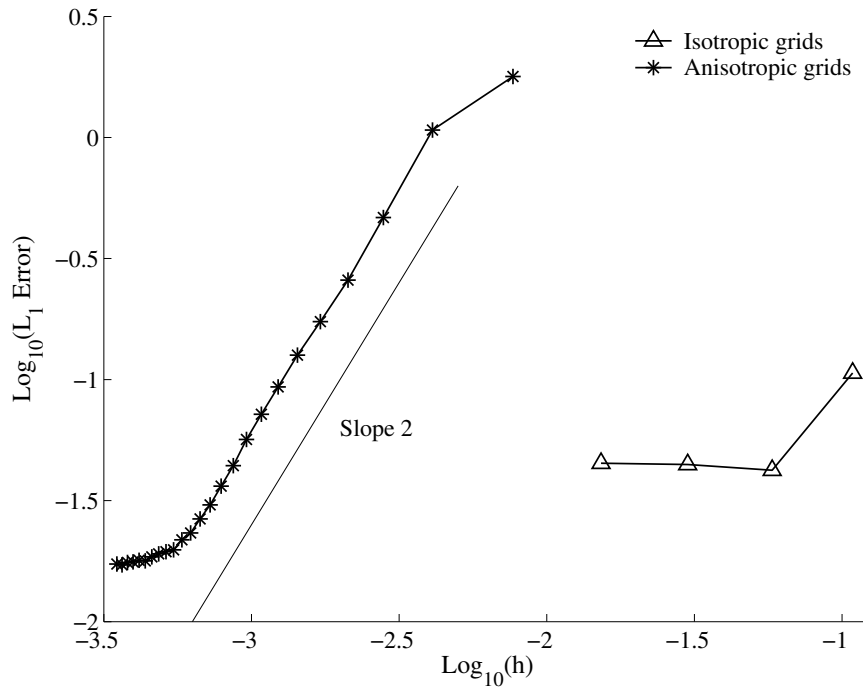


Figure E.3. Error convergence of the Galerkin scheme under positivity enforcement for the isotropic and anisotropic grid test cases.

the total edges, affecting 68 % of the total nodes. For the anisotropic case, again on all grids, the positivity enforcement was activated for 33 % of the total edges, affecting almost all nodes. See Figures E.4 and E.5. Also shown in the figures are the results for the Braaten-Connell scheme. The edge-coefficients were computed as described in Appendix D, and a node is considered as affected if the stencil centered at the node has at least one negative edge-coefficient. For the isotropic grids, the Braaten-Connell scheme has much less nodes affected (nearly 30 % of nodes are affected). For the anisotropic grids, similarly to the Galerkin scheme, almost all nodes have at least one negative coefficient. Despite the lack of positivity, the Braaten-Connell scheme produces surprisingly accurate solutions on these highly-skewed grids as shown in Figure 6.27.

Acknowledgments

The author would like to thank James L. Thomas, Pieter Buning, Mark Carpenter, Peter A. Gnoffo, William L. Kleb, Michael A. Park, Elizabeth M. Lee-Rausch, Charles C. Swanson, Veer N. Vatsa, Jeffery A. White (NASA Langley Research Center), and Boris Diskin (National Institute of Aerospace) for their constructive comments and suggestions. The author would like to thank also Yoshifumi Suzuki (Desktop Aeronautics) for his valuable comments. This work was partly supported by the NASA Fundamental Aeronautics Program through NASA Research Announcement Contract NNL07AA23C.

References

- ¹Gassner, G., Lörcher, F., and Munz, C. D., “A Contribution to the Construction of Diffusion Fluxes for Finite Volume and Discontinuous Galerkin schemes,” *Journal of Computational Physics*, Vol. 224, 2007, pp. 1049–1063.
- ²van Leer, B. and Lo, M., “Analysis and Implementation of Recovery-Based Discontinuous Galerkin for Diffusion,” *19th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2009-3786, San Antonio, 2009.
- ³Kannan, R., Sun, Y., and Wang, Z. J., “A Study of Viscous Flux Formulations for an Implicit P-Multigrid Spectral Volume Navier Stokes Solver,” *46th AIAA Aerospace Sciences Meeting*, AIAA Paper 2008-783, January 2008.
- ⁴Xu, Y. and Shu, C.-W., “Local Discontinuous Galerkin Methods for High-Order Time-Dependent Partial Differential Equations,” *Communications in Computational Physics*, Vol. 7, No. 1, 2010, pp. 1–46.
- ⁵Liu, H. and Yan, J., “The Direct Discontinuous Galerkin (DDG) Methods for Diffusion Problems,” *SIAM Journal of Numerical Analysis*, Vol. 47, No. 1, 2009, pp. 675–698.

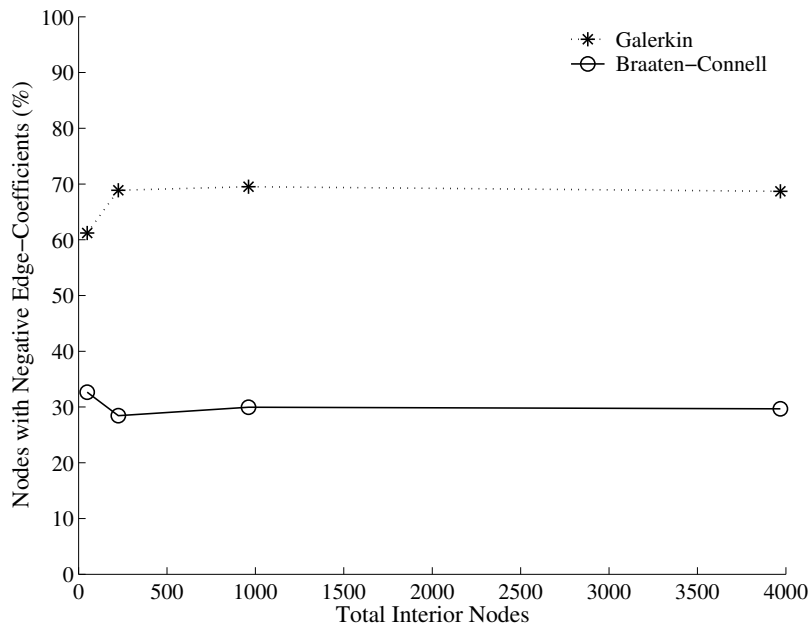


Figure E.4. Percentage of nodes having negative edge-coefficients for the Galerkin and Braaten-Connell schemes on isotropic grids.

⁶Peraire, J. and Persson, P.-O., “The Compact Discontinuous Galerkin (CDG) Method for Elliptic Problems,” *SIAM Journal of Scientific Computing*, Vol. 30, No. 2, 2008, pp. 1806–1824.

⁷Huynh, H. T., “A Reconstruction Approach to High-Order Schemes Including Discontinuous Galerkin for Diffusion,” *47th AIAA Aerospace Sciences Meeting*, AIAA Paper 2009-403, Orlando, 2009.

⁸Puigt, G., Auffray, V., and Müller, J.-D., “Discretization of Diffusive Fluxes on Hybrid Grids,” *Journal of Computational Physics*, Vol. 229, 2010, pp. 1425–1447.

⁹Nishikawa, H., “Towards Future Navier-Stokes Schemes: Uniform Accuracy, $O(h)$ time step, and Accurate Viscous/Heat Fluxes,” *19th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2009-3648, San Antonio, 2009.

¹⁰Veluri, S. P., Roy, C. J., Choudhary, A., and Luke, E. A., “Finite Volume Diffusion Operators for Compressible CFD on Unstructured Grids,” *19th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2009-4141, San Antonio, 2009.

¹¹Diskin, B., Thomas, J. L., Nielsen, E. J., Nishikawa, H., and White, J. A., “Comparison of Node-Centered and Cell-Centered Unstructured Finite-Volume Discretizations: Viscous Fluxes,” *AIAA Journal*, Vol. 48, No. 7, July 2010, pp. 1326–1338.

¹²Lipnikov, K., Svyatskiy, D., and Vassilevski, Y., “Interpolation-free monotone finite volume method for diffusion equations on polygonal meshes,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 703–716.

¹³Hermeline, F., “A finite volume method for approximating 3D diffusion operators on general meshes,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 5763–5786.

¹⁴Traore, P., Ahipo, Y. M., and Louste, C., “A robust and efficient finite volume scheme for the discretization of diffusive flux on extremely skewed meshes in complex geometries,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 5148–5159.

¹⁵Hermeline, F., “Monotone finite volume schemes for diffusion equations on polygonal meshes,” *Journal of Computational Physics*, Vol. 227, 2008, pp. 6288–6312.

¹⁶Breil, J. and Maire, P.-H., “A cell-centered diffusion scheme on two-dimensional unstructured meshes,” *Journal of Computational Physics*, Vol. 224, 2007, pp. 785–823.

¹⁷Cockburn, B. and Shu, C.-W., “Runge-Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems,” *Journal of Scientific Computing*, Vol. 16, No. 3, 2001, pp. 173–261.

¹⁸Sun, Y. and Wang, Z. J., “Formulations and Analysis of the Spectral Volume Method for the Diffusion Equation,” *Communications in Numerical Methods in Engineering*, Vol. 20, 2004, pp. 927–937.

¹⁹Trottenberg, U., Oosterlee, C. W., and Schüller, A., *Multigrid*, Academic Press, 2000.

²⁰Diskin, B. and Thomas, J. L., “Accuracy Analysis for Mixed-Element Finite-Volume Discretization Schemes,” *NIA Report No. 2007-08*, 2007.

²¹De Palma, P., Pascasio, G., Rubino, D. T., and Napolitano, M., “Residual Distribution Schemes for Advection and Advection-Diffusion Problems on Quadrilateral Cells,” *Journal of Computational Physics*, Vol. 218, 2006, pp. 159–199.

²²Arnold, D. N., “An Interior Penalty Finite Element Method with Discontinuous Elements,” *SIAM Journal on Numerical Analysis*, Vol. 19, No. 4, 1982, pp. 742–760.

²³Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, L. D., “Unified analysis of discontinuous Galerkin methods for elliptic problems,” *SIAM Journal on Numerical Analysis*, Vol. 39, No. 5, 2002, pp. 1749–1779.

²⁴Haselbacher, A., McGuirk, J. J., and Page, G. J., “Finite Volume Discretization Aspects for Viscous Flows on Mixed Unstructured Grids,” *AIAA Journal*, Vol. 37, No. 2, 1999, pp. 177–184.

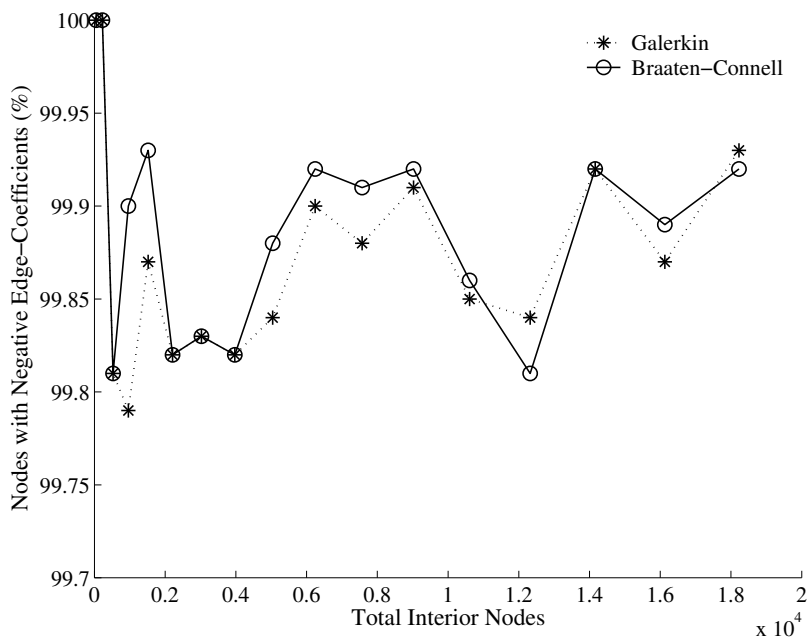


Figure E.5. Percentage of nodes having negative edge-coefficients for the Galerkin and Braaten-Connell schemes on anisotropic grids.

²⁵Weiss, J. M., Maruszkeski, J. P., and Smith, W. A., "Implicit Solution of Preconditioned Navier-Stokes Equations Using Algebraic Multigrid," *AIAA Journal*, Vol. 37, No. 1, 1999, pp. 29–36.

²⁶Nishikawa, H. and Roe, P. L., "On High-Order Fluctuation-Splitting Schemes for Navier-Stokes Equations," *Computational Fluid Dynamics 2004*, edited by C. Groth and D. W. Zingg, Springer-Verlag, 2004, pp. 799–804.

²⁷Ricchiuto, M., Villedieu, N., Abgrall, R., and Deconinck, H., "On Uniformly High-Order Accurate Residual Distribution Schemes for Advection-Diffusion," *Journal of Computational and Applied Mathematics*, Vol. 215, 2008, pp. 547–556.

²⁸van Leer, B. and Nomura, S., "Discontinuous Galerkin for Diffusion," *17th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2005-5108, Toronto, 2005.

²⁹Luo, H., Luo, L., Nourgaliev, R., and Mousseau, V. A., "A Reconstructed Discontinuous Galerkin Method for the Compressible Navier-Stokes Equations on Arbitrary Grids," *47th AIAA Aerospace Sciences Meeting*, AIAA Paper 2010-364, Orlando, 2010.

³⁰Cattaneo, C., "A Form of Heat-Conduction Equations which Eliminates the Paradox of Instantaneous Propagation," *Ct. R. Acad. Sci., Paris*, Vol. 247, 1958, pp. 431–433.

³¹Roe, P. L. and Arora, M., "Characteristic-Based Schemes for Dispersive Waves I. The Method of Characteristics for Smooth Solutions," *Numerical Methods for Partial Differential Equations*, Vol. 9, 1993, pp. 459–505.

³²Nishikawa, H., "A First-Order System Approach for Diffusion Equation. I: Second-Order Residual-Distribution Schemes," *Journal of Computational Physics*, Vol. 227, 2007, pp. 315–352.

³³Bassi, F. and Rebay, S., "A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations," *Journal of Computational Physics*, Vol. 131, 1997, pp. 267–279.

³⁴Nishikawa, H., "A First-Order System Approach for Diffusion Equation. II: Unification of Advection and Diffusion," *Journal of Computational Physics*, Vol. 229, 2010, pp. 3989–4016.

³⁵LeVeque, R. J., *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, 2002.

³⁶Tannehill, J. C., Anderson, D. A., and Pletcher, R. H., *Computational Fluid Mechanics and Heat Transfer*, Taylor & Francis, 2nd ed., 1997.

³⁷Godlewski, E. and Raviart, P.-A., *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, Springer, 1996.

³⁸Hirsch, C., *Numerical Computation of Internal and External Flows*, Vol. 2, A Wiley - Interscience Publications, 1990.

³⁹Rusanov, V. V., "Calculation of Interaction of Non-Steady Shock Waves with Obstacles," *J. Compt. Math. Phys. USSR*, Vol. 1, 1961, pp. 267–279.

⁴⁰Lax, P. D., "Weak Solutions of Nonlinear Hyperbolic Equations and Their Numerical Computation," *Communications on Pure and Applied Mathematics*, Vol. 7, 1954, pp. 159–193.

⁴¹Lax, P. and Wendroff, B., "Systems of Conservation Laws," *Communications on Pure and Applied Mathematics*, Vol. 13, 1960, pp. 217–237.

⁴²van Leer, B., "Towards the Ultimate Conservative Difference Scheme. V. A Second-Order Sequel to Godunov's Method," *Journal of Computational Physics*, Vol. 32, 1979, pp. 101–136.

⁴³Deconinck, H. and Abgrall, R., "Introduction to Residual Distribution Methods," *34th VKI CFD Lecture Series Very-High Order Discretization Methods*, VKI Lecture Series, 2005.

⁴⁴Wood, W. A., "Equivalence of Fluctuation Splitting and Finite Volume for One-Dimensional Gas Dynamics," NASA TM-97-206271, 1997.

- ⁴⁵Baumann, C. E., *An hp-adaptive Discontinuous Finite Element Method for Computational Fluid Dynamics*, Ph.D. thesis, University of Texas at Austin, 1997.
- ⁴⁶Oden, J. T., Babuška, I., and Baumann, C. E., “A Discontinuous hp Finite Element Method for Diffusion Problems,” *Journal of Computational Physics*, Vol. 146, 1998, pp. 491–519.
- ⁴⁷Shahbazi, K., “An Explicit Expression for the Penalty Parameter of the Interior Penalty Method,” *Journal of Computational Physics*, Vol. 205, 2005, pp. 401–407.
- ⁴⁸Hartmann, R. and Houston, P., “An Optimal Order Interior Penalty Discontinuous Galerkin Discretization of the Compressible Navier-Stokes Equations,” *Journal of Computational Physics*, Vol. 227, 2008, pp. 9670–9685.
- ⁴⁹Zhang, M. and Shu, C.-W., “An Analysis of Three Different Formulations of the Discontinuous Galerkin Method for Diffusion Equations,” *Mathematical Models and Methods in Applied Sciences*, Vol. 13, 2003, pp. 395–413.
- ⁵⁰van Raalte, M. H., *Multigrid Analysis and Embedded Boundary Conditions for Discontinuous Galerkin Discretization*, Ph.D. thesis, University of Amsterdam, 2004.
- ⁵¹Wang, Z. J., “Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids,” *Journal of Computational Physics*, Vol. 178, 2002, pp. 210–251.
- ⁵²Aftosmis, M., Gaitonde, D., and Tavares, T. S., “Behavior of Linear Reconstruction Techniques on Unstructured Meshes,” *AIAA Journal*, Vol. 33, No. 11, 1995, pp. 2038–2049.
- ⁵³Luo, H., Baum, J. D., and Löwner, R., “An Improved Finite Volume Scheme for Compressible Flows on Unstructured Grids,” *33rd Aerospace Sciences Meeting and Exhibit*, AIAA Paper 95-0348, 1995.
- ⁵⁴Haselbacher, A. and Blazek, J., “Accurate and Efficient Discretization of Navier-Stokes Equations on Mixed Grids,” *AIAA Journal*, Vol. 38, No. 11, 2000, pp. 2094–2102.
- ⁵⁵Thomas, J. L., Private Communication, 2009.
- ⁵⁶Thomas, J. L., Diskin, B., and Nishikawa, H., “A Critical Study of Agglomerated Multigrid Methods for Diffusion on Highly-Stretched Grids,” *Computers and Fluids*, 2010, to appear.
- ⁵⁷Braaten, M. E. and Connel, S. D., “Three-Dimensional Unstructured Adaptive Multigrid Scheme for the Navier-Stokes Equations,” *AIAA Journal*, Vol. 34, No. 2, February 1996, pp. 281–290.
- ⁵⁸Abgrall, R. and Mezine, M., “Construction of second order accurate monotone and stable residual distribution schemes for unsteady flow problems,” *Journal of Computational Physics*, Vol. 188, 2002, pp. 16–55.
- ⁵⁹Ricchiuto, M., Abgrall, R., and Deconinck, H., “Construction of Very High Order Residual Distribution Schemes for Unsteady Scalar Advection: Preliminary Results,” *33rd Computational Fluid Dynamics—Novel Methods for Solving Convection Dominated Systems*, VKI Lecture Series, May 2003.
- ⁶⁰Ricchiuto, M. and Abgrall, R., “Explicit Runge-Kutta Residual Distribution schemes for Time Dependent Problems: Second order case,” *Journal of Computational Physics*, Vol. 229, 2010, pp. 5653–5691.
- ⁶¹Mesaros, L. M. and Roe, P. L., “Multidimensional Fluctuation Splitting Schemes Based on Decomposition Methods,” *12th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 95-1699, San Diego, 1995.
- ⁶²Bastin, J. and Rogé, G., “A Multidimensional Fluctuation Splitting Scheme for the Three Dimensional Euler Equations,” *Mathematical Modelling and Numerical Analysis*, Vol. 33, 1999, pp. 1241–1259.
- ⁶³Diskin, B. and Thomas, J. L., “Accuracy of Gradient Reconstruction on Grids with High Aspect Ratio,” *NIA Report No. 2008-12*, 2008.
- ⁶⁴Nishikawa, H., Diskin, B., and Thomas, J. L., “Critical Study of Agglomerated Multigrid Methods for Diffusion,” *AIAA Journal*, Vol. 48, No. 4, April 2010, pp. 839–847.
- ⁶⁵Mavriplis, D. J., “Multigrid Techniques for Unstructured Meshes,” *VKI Lecture Series VKI-LS 1995-02, Von Karman Institute for Fluid Dynamics, Rhode-Saint-Genese, Belgium*, 1995.
- ⁶⁶Nishikawa, H., Diskin, B., and Thomas, J. L., “Development and Application of Agglomerated Multigrid Methods for Complex Geometries,” *40th AIAA Fluid Dynamics Conference and Exhibit*, AIAA Paper 2010-4731, Chicago, 2010.
- ⁶⁷Cockburn, B. and Shu, C. W., “The Runge-Kutta Discontinuous Galerkin Method for Conservation Laws: V. Multidimensional Systems,” *Journal of Computational Physics*, Vol. 141, 1998, pp. 199–224.
- ⁶⁸Atkins, H. L. and Shu, C.-W., “Analysis of the Discontinuous Galerkin Method Applied to the Diffusion Operator,” *14th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 99-3306, Norfolk, 1999.
- ⁶⁹Shahbazi, K., Mavriplis, D., and Burgess, N. K., “Multigrid Algorithms for High-Order Discontinuous Galerkin Discretizations of the Compressible Navier-Stokes Equations,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 7917–7940.
- ⁷⁰Wang, Z. J. and Liu, Y., “Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids: II. Extension to Two-Dimensional Scalar Equation,” *Journal of Computational Physics*, Vol. 179, 2002, pp. 665–697.
- ⁷¹Zienkiewicz, O. C. and Taylor, R. L., *The Finite Element Method*, Volume 1, McGraw-Hill Company, 1994.
- ⁷²Shahbazi, K., Fischer, P. F., and Ethier, C. R., “A High-Order Discontinuous Galerkin Method for the Unsteady Incompressible Navier-Stokes Equations,” *Journal of Computational Physics*, Vol. 222, 2007, pp. 391–407.
- ⁷³Mavriplis, D. J., “Revisiting the Least-Squares Procedure for Gradient Reconstruction on Unstructured Meshes,” *16th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2003-3986, Orlando, 2003.
- ⁷⁴Barth, T. J., “Recent Developments in High Order K-Exact Reconstruction on Unstructured Meshes,” AIAA Paper 93-0668, 1993.
- ⁷⁵Barth, T. J., “Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes,” AIAA Paper 91-0721, 1991.
- ⁷⁶Gnoffo, P. A., “Multi-Dimensional, Inviscid Flux Reconstruction for Simulation of Hypersonic Heating on Tetrahedral Grids,” AIAA Paper 2009-599, 2009.
- ⁷⁷Nishikawa, H., “Higher-Order Discretization of Diffusion Terms in Residual-Distribution Methods,” *34th VKI CFD Lecture Series Very-High Order Discretization Methods*, VKI Lecture Series, 2005.
- ⁷⁸Crumpton, P. I., Mackenzie, J. A., and Morton, K. W., “Cell Vertex Algorithms for the Compressible Navier-Stokes Equations,” *Journal of Computational Physics*, Vol. 109, 1993, pp. 1–15.

⁷⁹Liu, Y., Vinokur, M., and Wang, Z. J., "Spectral Difference Method for Unstructured Grids I. Basic Formulation," *Journal of Computational Physics*, Vol. 216, 2006, pp. 780–801.

⁸⁰Corre, C., Hanss, G., and Lerat, A., "A Residual-Based Compact Schemes for the Unsteady Compressible Navier-Stokes Equations," *Computers and Fluids*, Vol. 34, 2005, pp. 561–580.

⁸¹Katz, A. and Jameson, A., "A Multi-Solver Scheme for Viscous Flows Using Adaptive Cartesian Grids and Meshless Grid Communication," *47th AIAA Aerospace Sciences Meeting*, AIAA Paper 2009-768, January 2009.

⁸²Abgrall, R. and Roe, P. L., "High-Order Fluctuation Schemes on Triangular Meshes," *Journal of Scientific Computing*, Vol. 19, No. 1-3, 2002, pp. 3–36.

⁸³Frink, N. T., "Tetrahedral Unstructured Navier-Stokes Method for Turbulent Flows," *AIAA Journal*, Vol. 36, No. 11, 1998, pp. 1975–1982.

As soon as tradition has come to be recognized as tradition, it is dead.

Allan Bloom