

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/269248010>

Truncation and Discretization Error for Diffusion Schemes on Unstructured Meshes

Conference Paper · January 2014

DOI: 10.2514/6.2014-0478

CITATIONS

3

READS

193

4 authors:



[Gary Yan](#)

University of British Columbia - Vancouver

10 PUBLICATIONS 79 CITATIONS

[SEE PROFILE](#)



[Varun Puneria](#)

University of British Columbia - Vancouver

2 PUBLICATIONS 3 CITATIONS

[SEE PROFILE](#)



[Alireza Jalali](#)

University of British Columbia - Vancouver

18 PUBLICATIONS 185 CITATIONS

[SEE PROFILE](#)



[Carl Frederick Ollivier-Gooch](#)

University of British Columbia - Vancouver

159 PUBLICATIONS 2,809 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Shock Fitting on Unstructured Meshes [View project](#)



hp-adaptation for unstructured finite volume methods [View project](#)

Truncation and Discretization Error for Diffusion Schemes on Unstructured Meshes

Gary Kai Kin Yan*
Varun Prakash Puneria†
Alireza Jalali‡
Carl Ollivier-Gooch§

Department of Mechanical Engineering
The University of British Columbia
Vancouver, BC, Canada V6T 1Z4

I. Introduction

The accuracy and reliability of CFD simulations depend on the ability to reduce and quantify physical modeling and numerical errors. The fact that numerical errors are at least as large as physical modeling errors was highlighted in the results of the 3rd AIAA Drag Prediction workshop, which also showed the increased severity of this issue for unstructured flow solvers.¹ Furthermore, the varied local shape and connectivity of unstructured meshes make it difficult to quantify numerical error. The 5th AIAA Drag Prediction workshop² focussed on reducing grid-related errors even further, where a grid refinement study was performed on a common grid sequence derived from a multiblock structured grid. The study had six different levels of grid refinement ranging from 136×10^6 cells to 0.64×10^6 cells, a much larger range than is typically seen, with structured overset and hexahedral, prismatic, tetrahedral, and hybrid unstructured grid formats. The results of the grid refinement study indicated that there was no clear advantage of any one grid type in terms of a reduced scatter in solution. Moreover, there were no clear breakouts with grid type or turbulence model. The conclusion was that discretization errors and turbulence modeling errors are both still major contributors to error in solution. The impact on solution accuracy by the interactions between mesh quality (cell size, shape, and anisotropy) and discretization schemes is not well understood and demands further investigation.

The difference between the discrete operator and the continuous PDE applied to the solution is referred to as the truncation error, while the difference between the numerically approximated solution and the exact solution is the discretization error. The truncation error can be expressed in terms of the derivatives of an underlying smooth solution at the points of the discrete domain and can be used to estimate the discretization errors that occur during the approximate numerical solution of PDEs. It can be shown³ that for the special case of a linear differential operator \mathcal{L} , the truncation error τ can be used to calculate the discretization error ε , as

$$\mathcal{L}(\varepsilon) = \tau, \quad (1)$$

known as the error transport equation. For a general nonlinear differential operator, Banks et al.⁴ noted that the error transport equation takes another form different from simply replacing \mathcal{L} by \mathcal{N} in Eq. (1). However, the scope of the current work is restricted to linear problems, where Eq. (1) governs the transport of discretization error through the domain. This equation gives a relationship between the two measures

*MASC Candidate, gary.yan@alumni.ubc.ca

†MASC Candidate, puneria.vp@alumni.ubc.ca

‡PhD Candidate, arjalali@interchange.ubc.ca

§Professor, cfog@mech.ubc.ca, Member AIAA

of error, with truncation error being the source that drives discretization error in the solution. It has been observed⁵ that for at least some cases, refinement of the mesh where the truncation error is high gives lower discretization error in the solution, motivating investigation of the possible relationship between these two notions of error.

Given the significance of truncation error in the error transport equation, it is insightful to characterize and quantify both discretization error as well as truncation error for different discretization schemes. The accuracy assessment is carried out on Poisson's equation, and the results may guide us in the choice of discretization schemes for more complex governing equations. In solving Poisson's equation as a model of viscous flow using the finite-volume approach, Diskin et al.⁶ compared cell-centered and vertex-centered schemes and concluded that there is little difference in the accuracy of the two schemes at an equivalent number of degrees of freedom. The goal of the current research is to quantify truncation error and discretization error for unstructured finite-volume solvers using the vertex-centered approach.

We begin with performing analytic tests on structured triangular meshes (regular topology) with different classes of distortions to the geometry. The approach here is similar to that used by Jalali and Ollivier-Gooch,⁷ whose work involved the computation of the coefficients in a series expansion of truncation error for cell-centered schemes. For the general triangular unstructured mesh case, we perform a truncation error and discretization error analysis using an unstructured finite-volume solver. The paper is organized as follows. In section II, the discretization schemes are discussed with an emphasis on the vertex-centered approach. The calculation of gradients by the least-squares method with different flux averaging schemes and the Green-Gauss method is described. The use of a jump term to address the discontinuity of reconstructed solutions and gradients across faces is also explained. Section III describes our approaches in calculating the discretization error and truncation error, with results summarized in section IV. Overall concluding remarks are summarized in section V.

II. Discretization Schemes

To compute the approximate solution to Laplace's Equation, $\nabla^2 \phi = 0$, we cast it in the finite-volume formulation by integrating over a control volume Ω_i and applying the divergence theorem to obtain

$$\oint_{\partial\Omega_i} \nabla \phi \cdot \hat{\mathbf{n}} ds = 0. \quad (2)$$

The inclusion of a source term is omitted in the derivation, but it can be added in a straightforward manner. The spatial discretization of control volumes Ω_i and the computation of the flux $\nabla \phi$ at the boundaries of the control volumes are required to compute Eq. (2) numerically.

II.A. Vertex-Centered Schemes

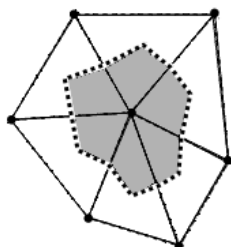


Figure 1: A vertex-centered control volume. The solid line segments are the primal faces of the mesh, and the dotted line segments are the dual faces of the shaded control volume.

In the vertex-centered spatial discretization, the mesh of triangular primal cells is first constructed. The control volumes are then formed using the median dual partition. There is one control volume per vertex, with bounding dual faces formed by connecting cell centroids and primal face midpoints. For a vertex with

N vertex neighbors, the control volume will be a $2N$ -gon containing the original vertex. This discretization generates a set of non-overlapping control volumes that fills the domain. For each control volume, the reference point is taken to be at the vertex. However, the centroid of the control volume is in general not located at the vertex, and the consequences of this subtlety is further discussed in later sections. A vertex-centered control volume along with the primal triangular mesh is shown in figure 1. This vertex-centered discretization is in contrast to the cell-centered control volume discretization, where each triangular primal cell is taken to be a control volume.

II.B. Least-squares Reconstruction

The following is a discussion on the methods by which the fluxes are computed using least-squares reconstruction. A detailed discussion is presented in the work of Barth and Frederickson,⁸ and Ollivier-Gooch and van Altena.⁹ Some major aspects are highlighted here.

II.B.1. Control Volume Gradient

The flux integral expression in Eq. (2) requires the computation of the gradients of solution at the control volume boundaries. This is performed by first reconstructing a polynomial within each control volume using solution average information from the considered control volume and nearby control volumes. The reconstructed polynomial $\Phi_i(x, y)$ within control volume i is a Taylor Series expansion having the form

$$\begin{aligned}\Phi_i(x, y) = & \phi|_i + \frac{\partial\phi}{\partial x}\Big|_i (x - x_i) + \frac{\partial\phi}{\partial y}\Big|_i (y - y_i) + \\ & \frac{1}{2} \frac{\partial^2\phi}{\partial x^2}\Big|_i (x - x_i)^2 + \frac{\partial^2\phi}{\partial x\partial y}\Big|_i (x - x_i)(y - y_i) + \frac{1}{2} \frac{\partial^2\phi}{\partial y^2}\Big|_i (y - y_i)^2 + \dots\end{aligned}\quad (3)$$

where the subscript i indicates the quantity at the reference point of control volume i . Enforcement of the mean in control volume i gives

$$\frac{1}{A_i} \int_{\Omega_i} \Phi_i(x, y) dA = \bar{\phi}_i. \quad (4)$$

Substituting Eq. (3) then yields

$$\bar{\phi}_i = \frac{1}{A_i} \int_{\Omega_i} \Phi_i(x, y) dA = \phi|_i + \frac{\partial\phi}{\partial x}\Big|_i \bar{x}_i + \frac{\partial\phi}{\partial y}\Big|_i \bar{y}_i + \frac{1}{2} \frac{\partial^2\phi}{\partial x^2}\Big|_i \bar{x}_i^2 + \frac{\partial^2\phi}{\partial x\partial y}\Big|_i \bar{x}_i \bar{y}_i + \frac{1}{2} \frac{\partial^2\phi}{\partial y^2}\Big|_i \bar{y}_i^2 + \dots \quad (5)$$

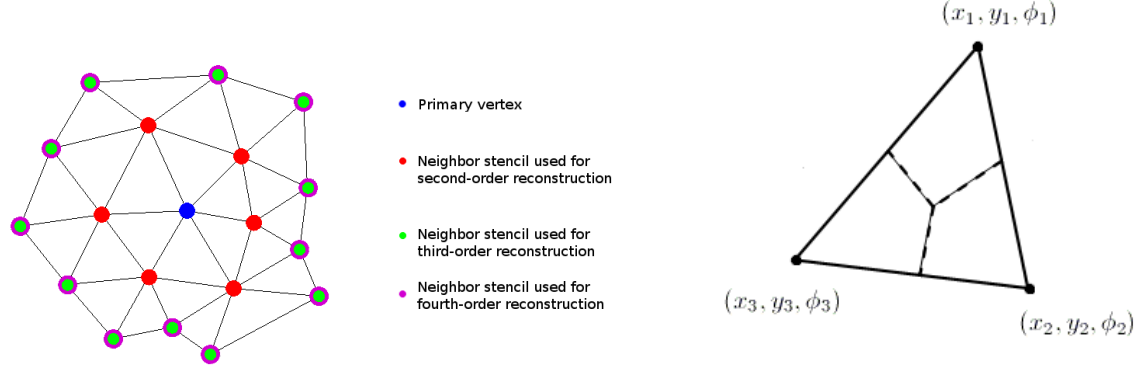
where the purely geometric control volume moments are defined as

$$\overline{x^p y^q}_i := \frac{1}{A_i} \int_{\Omega_i} (x - x_i)^p (y - y_i)^q dA. \quad (6)$$

The unknown coefficients $\left\{ \phi|_i, \frac{\partial\phi}{\partial x}\Big|_i, \frac{\partial\phi}{\partial y}\Big|_i, \dots \right\}$ in Eq. (3) need to be determined for the reconstructed polynomial. This is done by applying conditions that try to enforce the mean in nearby control volumes in the stencil of Ω_i ,

$$\begin{aligned}\frac{1}{A_j} \int_{\Omega_j} \Phi_i(x, y) dA = & \phi|_i + \frac{\partial\phi}{\partial x}\Big|_i \frac{1}{A_j} \int_{\Omega_j} (x - x_i) dA + \frac{\partial\phi}{\partial y}\Big|_i \frac{1}{A_j} \int_{\Omega_j} (y - y_i) dA + \\ & \frac{1}{2} \frac{\partial^2\phi}{\partial x^2}\Big|_i \frac{1}{A_j} \int_{\Omega_j} (x - x_i)^2 dA + \frac{\partial^2\phi}{\partial x\partial y}\Big|_i \frac{1}{A_j} \int_{\Omega_j} (x - x_i)(y - y_i) dA + \\ & \frac{1}{2} \frac{\partial^2\phi}{\partial y^2}\Big|_i \frac{1}{A_j} \int_{\Omega_j} (y - y_i)^2 dA + \dots\end{aligned}\quad (7)$$

The number of neighbors in the stencil required for a given order of accuracy is shown in figure 2(a).



(a) Stencils for different orders of accuracy for the vertex in blue, added one layer at a time, used for the least squares schemes. The red indicates the neighbor vertices needed for second order reconstruction, and green and purple for third and fourth order. They coincide because adding in all the second neighbors is sufficient for both orders.

(b) Green-Gauss gradient of dual faces within a primal cell.

Figure 2: The stencils required for reconstruction in least squares schemes, compared with computation of the Green-Gauss gradient.

To simplify the computation, $x - x_i$ and $y - y_i$ are replaced with $(x - x_j) + (x_j - x_i)$ and $(y - y_j) + (y_j - y_i)$ respectively, which gives

$$\begin{aligned}
 \frac{1}{A_j} \int_{\Omega_j} \Phi_i(x, y) dA &= \phi|_i + \frac{\partial \phi}{\partial x} \Big|_i (\bar{x}_j + (x_j - x_i)) + \frac{\partial \phi}{\partial y} \Big|_i (\bar{y}_j + (y_j - y_i)) + \\
 &\quad \frac{1}{2} \frac{\partial^2 \phi}{\partial x^2} \Big|_i (\bar{x}_j^2 + 2\bar{x}_j(x_j - x_i) + (x_j - x_i)^2) + \\
 &\quad \frac{\partial^2 \phi}{\partial x \partial y} \Big|_i (\bar{x}_j \bar{y}_j + \bar{x}_j(y_j - y_i) + \bar{y}_j(x_j - x_i) + (x_j - x_i)(y_j - y_i)) + \\
 &\quad \frac{1}{2} \frac{\partial^2 \phi}{\partial y^2} \Big|_i (\bar{y}_j^2 + 2\bar{y}_j(y_j - y_i) + (y_j - y_i)^2) + \dots
 \end{aligned} \tag{8}$$

Assuming six first vertex neighbors in the stencil for second order reconstruction are used, combining the above will lead to an overdetermined system of linear algebraic equations

$$\begin{bmatrix}
 w_{i1}(\bar{x}_1 - \bar{x}_i + x_1 - x_i) & w_{i1}(\bar{y}_1 - \bar{y}_i + y_1 - y_i) \\
 w_{i2}(\bar{x}_2 - \bar{x}_i + x_2 - x_i) & w_{i2}(\bar{y}_2 - \bar{y}_i + y_2 - y_i) \\
 w_{i3}(\bar{x}_3 - \bar{x}_i + x_3 - x_i) & w_{i3}(\bar{y}_3 - \bar{y}_i + y_3 - y_i) \\
 w_{i4}(\bar{x}_4 - \bar{x}_i + x_4 - x_i) & w_{i4}(\bar{y}_4 - \bar{y}_i + y_4 - y_i) \\
 w_{i5}(\bar{x}_5 - \bar{x}_i + x_5 - x_i) & w_{i5}(\bar{y}_5 - \bar{y}_i + y_5 - y_i) \\
 w_{i6}(\bar{x}_6 - \bar{x}_i + x_6 - x_i) & w_{i6}(\bar{y}_6 - \bar{y}_i + y_6 - y_i)
 \end{bmatrix}
 \begin{bmatrix} \frac{\partial \phi}{\partial x} \Big|_i \\ \frac{\partial \phi}{\partial y} \Big|_i \end{bmatrix} = \begin{bmatrix} w_{i1}(\bar{\phi}_1 - \bar{\phi}_i) \\ w_{i2}(\bar{\phi}_2 - \bar{\phi}_i) \\ w_{i3}(\bar{\phi}_3 - \bar{\phi}_i) \\ w_{i4}(\bar{\phi}_4 - \bar{\phi}_i) \\ w_{i5}(\bar{\phi}_5 - \bar{\phi}_i) \\ w_{i6}(\bar{\phi}_6 - \bar{\phi}_i) \end{bmatrix}. \tag{9}$$

Each row is multiplied by a weight w_{ij} to emphasize control volumes with reference points closer in the sense of the Euclidean norm to the reference point of the control volume in question, given by

$$w_{ij} = \frac{1}{|\mathbf{r}_j - \mathbf{r}_i|}, \tag{10}$$

where \mathbf{r}_j is the position vector of the reference point of control volume j in the stencil and \mathbf{r}_i is the position vector of the considered control volume i . Each interior vertex will have at least three vertex neighbors, so an overdetermined system, as Eq. (9), will be obtained. This least-squares system is solved to determine the gradient of the control volume to be used as coefficients of the reconstructed polynomial in Eq. (3).

II.B.2. Face Gradient - Flux Averaging Schemes

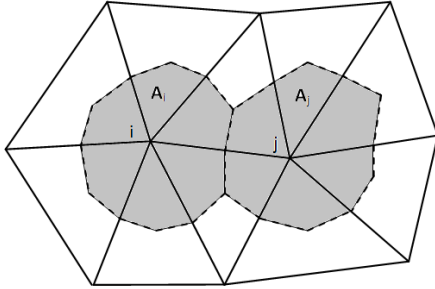
To evaluate Eq. (2), integration of the gradient around the boundary of the control volumes is needed. This is done to the desired order of accuracy using Gauss quadrature. For second order accuracy, one Gauss point is needed at the midpoint of each dual face. Once the control volume gradients are determined, evaluation of the reconstructed solution, Eq. (3), at the Gauss points is straightforward. However, this value of the solution and gradient at the midpoint of the dual face will in general be different for the two control volumes that share that dual face, because Eq. (3) is a piecewise polynomial for each control volume. One simple method of resolving this discontinuity across each dual face is to take an arithmetic average of the flux on either side of the face, denoted by the superscripts L and R , as

$$\nabla\phi = \frac{1}{2}(\nabla\phi^L + \nabla\phi^R). \quad (11)$$

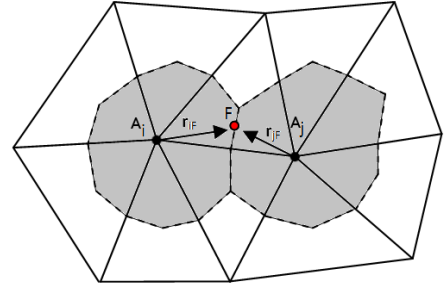
This method, referred to as Arithmetic Averaging, is one of the simplest ways to incorporate flux information from both sides. We also consider two other methods of combining the flux from both sides, as was considered for the cell-centered case.⁷ The other two averaging schemes are Volume-Weighted and Linear Interpolation, whose expressions are given in table 1 and illustrations are given in figure 3. Intuitively, the Volume-Weighted averaging emphasizes the gradient from the control volume that has a larger area, while Linear Interpolation emphasizes the gradient from the closer control volume.

Scheme	Formula
Arithmetic Averaging (AA)	$\frac{1}{2}\nabla\phi_i + \frac{1}{2}\nabla\phi_j$
Volume-Weighted (VW)	$\frac{A_i}{A_i+A_j}\nabla\phi_i + \frac{A_j}{A_i+A_j}\nabla\phi_j$
Linear Interpolation (LI)	$\frac{ \mathbf{r}_{jF} }{ \mathbf{r}_{iF} + \mathbf{r}_{jF} }\nabla\phi_i + \frac{ \mathbf{r}_{iF} }{ \mathbf{r}_{iF} + \mathbf{r}_{jF} }\nabla\phi_j$

Table 1: Computation of face gradient using averaging schemes.



(a) Volume-Weighted averaging scheme.



(b) Linear Interpolation averaging scheme.

Figure 3: Illustration of quantities used in table 1.

A method to alleviate the discontinuity across the dual face has been proposed by Nishikawa,¹⁰ known as the jump term, added to the averaged flux in the face normal direction, in the form of

$$\nabla\phi_{\text{Jump}} = \frac{\alpha}{|\mathbf{r}_{ij} \cdot \hat{\mathbf{n}}_F|}(\phi^R - \phi^L)\hat{\mathbf{n}}_F. \quad (12)$$

In this equation $\hat{\mathbf{n}}_F$ is the unit normal of the dual face and \mathbf{r}_{ij} is the vector connecting the reference point of control volume i to that of control volume j . The parameter α , known as the jump term coefficient, is used to characterize the application of the jump term.

II.C. Green-Gauss Reconstruction

In the Green-Gauss approach, as described by Diskin et al.,⁶ the gradient is computed at each dual face directly. For second order reconstruction, the solution gradient is piecewise constant. It is assumed that the

solution gradient is piecewise constant within each primal cell, and takes the value of the unique direction of steepest increase defined by the three control volume averages at the vertices. This value for the gradient is used when computing the flux integral for each dual face, so that

$$\nabla\phi_{\text{dual face}} = \nabla\phi_{\text{cell}}. \quad (13)$$

Using only triangular cells, this formulation is equivalent to a Galerkin finite-element scheme with linear basis functions.¹¹ If x_j, y_j , and ϕ_j , $j = 1, 2, 3$ represents the x -coordinate, y -coordinate, and solution values at the three vertices that define the primal cell, then one way to express the gradient within the primal cell is

$$\nabla\phi_{\text{cell}} = \begin{bmatrix} \frac{(\phi_2 - \phi_1)(y_3 - y_1) - (y_2 - y_1)(\phi_3 - \phi_1)}{(x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)} & \frac{(x_2 - x_1)(\phi_3 - \phi_1) - (\phi_2 - \phi_1)(x_3 - x_1)}{(x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)} \end{bmatrix}^T. \quad (14)$$

This is visualized in figure 2(b), compared with the stencils required for least squares reconstruction discussed earlier. All three dual faces within this control volume is assigned the unique gradient that is determined from the solution values at the vertices.

This approach implies a different interpretation of the value of the solution, that being the pointwise value at the vertex rather than the control volume average corresponding to that vertex. The distinction between pointwise values at vertices and control volume averages is due to the centroid of the control volume not coinciding with the vertex location.

It should be noted that in computing the flux in the Green-Gauss method, the gradient at each dual face takes on a single value although there are two control volumes that share it. This can be seen since each dual face is contained within a unique primal cell. Because of this property, there is no discontinuity across the face and no jump term is needed.

III. Methodology

III.A. Analytic Tests

The first class of meshes analyzed are those with regular topology. This means that for every interior vertex, there is exactly the same number of neighbors. For this analysis we assume that each vertex has six neighbors. The control volumes are constructed in the same way using the median dual partition, as in the approach taken by Jalali and Olliver-Gooch.⁷ A computer algebra system was used to compute the truncation error for different meshes in this category. In control volume form, the flux integral for Laplace's equation is

$$\mathcal{R}(\phi_i) = \widetilde{\nabla^2\phi_i} = \frac{1}{A_i} \oint_{\partial\Omega_i} \nabla\phi \cdot \hat{\mathbf{n}} ds. \quad (15)$$

This is discretely computed as a sum over all K dual faces, using the one-point Gauss quadrature rule

$$\mathcal{R}(\phi_i) \approx \frac{1}{A_i} \sum_K \nabla\phi^K \cdot \hat{\mathbf{n}}_K l_K \quad (16)$$

where $\nabla\phi^K$ is the gradient at the midpoint of face K and $\hat{\mathbf{n}}_K$ and l_K are the normal and length of face K , respectively. Using the definitions of moments in Eq. (6), the average of the exact flux integral is

$$\overline{\nabla^2\phi_i} = \left. \frac{\partial^2\phi}{\partial x^2} \right|_i + \left. \frac{\partial^2\phi}{\partial y^2} \right|_i + \left. \frac{\partial^3\phi}{\partial x^3} \right|_i \bar{x}_i + \left. \frac{\partial^3\phi}{\partial x^2\partial y} \right|_i \bar{y}_i + \left. \frac{\partial^3\phi}{\partial x\partial y^2} \right|_i \bar{x}_i + \left. \frac{\partial^3\phi}{\partial y^3} \right|_i \bar{y}_i + \dots \quad (17)$$

from which the truncation error is computed as

$$\tau_i = \widetilde{\nabla^2\phi_i} - \overline{\nabla^2\phi_i} = h^{-1} \left(\lambda_x \left. \frac{\partial\phi}{\partial x} \right|_i + \lambda_y \left. \frac{\partial\phi}{\partial y} \right|_i \right) + h^0 \left(\lambda_{xx} \left. \frac{\partial^2\phi}{\partial x^2} \right|_i + \lambda_{xy} \left. \frac{\partial^2\phi}{\partial x\partial y} \right|_i + \lambda_{yy} \left. \frac{\partial^2\phi}{\partial y^2} \right|_i \right) + \mathcal{O}(h), \quad (18)$$

where $\{\lambda_x, \lambda_y, \lambda_{xx}, \lambda_{xy}, \lambda_{yy}, \dots\}$ are the truncation error coefficients.

In general, for a control volume corresponding to a vertex with N neighbors, it can be divided into $2N$ triangles, by adding line segments connecting primal cell centroids to the vertex in question. For the regular

topology case with six vertex neighbors there will be twelve such triangles. In the least-squares method where control volume moments are needed about a reference point, the moments are first computed for each of the twelve triangles about its own centroid, and each contribution is transferred to moments about the control volume reference point via the parallel axis theorem, Eq. (8). Different geometries of the mesh are considered, as described below. The types of meshes are shown in figure 4.

III.A.1. Uniform Mesh

This is the baseline mesh with equilateral triangles of side length h .

III.A.2. Sheared Mesh

The y -coordinate of each vertex of the primal mesh is unaltered from the uniform mesh. For convenience, we will define an enumeration of rows for vertices, where the origin is in row 0 and increases upwards. All the vertices in the same row m with the same y -coordinate are translated horizontally by a constant amount $b m h$, where $b > 0$ is a constant shearing factor.

III.A.3. Scaled Mesh

The x -coordinate of each vertex of the primal mesh is unaltered from the uniform mesh. The y -coordinate of each vertex is multiplied by a constant scaling factor $k > 0$.

III.A.4. Curved Mesh

Vertices that lie on a horizontal line of the uniform mesh with the same y -coordinate are mapped onto a circular arc with the same radius r from the center. Vertices that lie on a vertical line of the uniform mesh with the same x -coordinate are mapped onto a radial line with the same polar angle θ .

III.A.5. Stretched Mesh

The x -coordinate of each vertex of the primal mesh is unaltered from the uniform mesh. A stretching factor s is defined to determine the height of each row of vertices. For $0 < s < 1$ the height of the triangle between rows $m + 1$ and m would be given by $\frac{\sqrt{3}}{2} h s^{-m}$.

III.A.6. Perturbed Mesh

The x and y -coordinates of each vertex are shifted independently by a factor $\rho_x h$ and $\rho_y h$ respectively, where $\rho_x, \rho_y \in [-0.1, 0.1]$ are uniformly distributed random numbers. The small range of this parameter ensures that the cells will not be overlapping.

III.B. Numerical Tests

III.B.1. Truncation Error

Next, tests were performed on general unstructured meshes. Here each vertex away from the boundary will have at least three vertex neighbors, and the number of neighbors will in general be different for different vertices in the mesh. The approach taken for the regular topology case with six neighbors considered earlier, using computer algebra software, cannot be used directly due to the stencil being different across the mesh. Instead, we exploit the fact that the discrete flux integral for linear equations can be written as a linear combination of the solution average in control volumes in the stencil S_i , as

$$\widetilde{\nabla^2 \phi}_i = \sum_{j \in S_i} \frac{\partial R_i}{\partial \phi_j} \bar{\phi}_j, \quad (19)$$

where $\frac{\partial R_i}{\partial \phi_j}$, $j \in S_i$ is one row of the global flux Jacobian. This is computed in the solver code explicitly, as explained in the work by Michalak.¹² Then Eq. (7) is used to express $\bar{\phi}_j$ as a series expansion of control volume averages in terms of solution and derivatives of control volume i . The terms are collected to obtain the truncation error. To compute the required moments for control volumes, the area integral is transformed

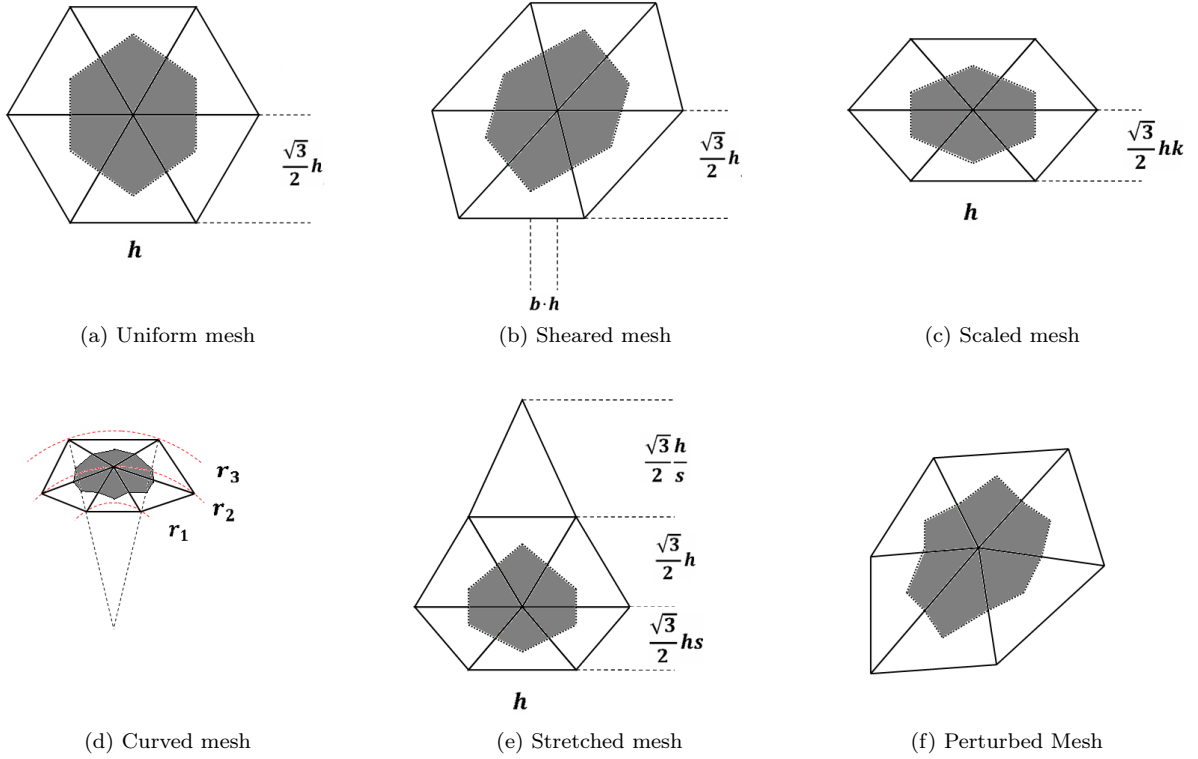


Figure 4: Uniform and distorted meshes.

into a line integral over the boundary. This is numerically integrated up to the desired order of accuracy using Gauss quadrature for each dual face.

III.B.2. Discretization Error

Having quantified the truncation error τ in the two classes of meshes, the relationship between with the discretization error $\epsilon = \tilde{\phi} - \phi$ is investigated. The Poisson problem will be solved using an unstructured finite-volume solver that is nominally second order. The converged solution is used to compute the discretization error in the control volume averages. The method of manufactured solution is employed, where the Laplacian operator ∇^2 is acted on the desired exact solution, and the result is used as the source term.

Some properties that are investigated include the effect of different flux averaging schemes, choice of jump term coefficient on reducing truncation and discretization error, and the extent to which correlations of reducing both types of error exist, as presented in the results. These two notions of error measure the accuracy of the computed solution, with different interpretations. On unstructured meshes it is known that in different cases the prescribed order of accuracy in the discretization error can be obtained even when the truncation error does not converge at that same order, or sometimes even does not converge at all.¹³

IV. Results

IV.A. Regular Topology - Analytic Tests

The truncation error of regular topology meshes was analyzed by computing Eq. (16) using a computer algebra system. For comparison, values for constants of the distorted meshes were chosen and summarized in table 2 along with the corresponding error coefficients in the form of Eq. (18). The results here are for the Arithmetic Averaging scheme. Some tests were done for the other schemes and found to yield similar results, though not presented here for brevity. The purpose here is a proof of concept for the analytic method, and we focus our attention on comparing the mentioned schemes for the general unstructured mesh case handled in the numerical tests.

Mesh	λ_{xxxx}	λ_{xxxy}	λ_{xxyy}	λ_{xyyy}	λ_{yyyy}
Least-Squares					
Uniform	0.2500	0	0.5000	0	0.2500
Sheared ($b = 0.2$)	0.2642	0.1216	0.4974	0.0881	0.2500
Scaled ($k = 0.8$)	0.2409	0	0.4276	0	0.1600
Green-Gauss					
Uniform	0.0625	0	0.1250	0	0.0625
Sheared ($b = 0.2$)	0.0686	0.0323	0.1050	0	0.0625
Scaled ($k = 0.8$)	0.0508	0	0.1250	0	0.0400

(a) Leading order truncation error coefficients for schemes with second order truncation error.

Mesh	λ_x	λ_y	λ_{xx}	λ_{xy}	λ_{yy}
Least-Squares					
Curved ($r_1 = 1$, $\frac{r_{i+1}}{r_i} = 1.15$)	0	0	-0.2482	0	-0.0602
Stretched	0	0	-0.0023	0	0.0274
Perturbed	0	0	-0.0298	0.0147	0.0142
Green-Gauss					
Curved ($r_1 = 1$, $\frac{r_{i+1}}{r_i} = 1.15$)	0	0	0.2851	0	-0.0932
Stretched	0	0	0	0	0.0308
Perturbed	-0.0107	-0.1729	-0.0737	0.1353	-0.0110

(b) Leading order truncation error coefficients for schemes with lower order truncation error.

Table 2: Truncation error coefficients for analytic tests.

Mesh	Uniform	Sheared	Scaled	Curved	Stretched	Perturbed
Least-Squares	$\mathcal{O}(h^2)$	$\mathcal{O}(h^2)$	$\mathcal{O}(h^2)$	$\mathcal{O}(h^0)$	$\mathcal{O}(h^0)$	$\mathcal{O}(h^0)$
Green-Gauss	$\mathcal{O}(h^2)$	$\mathcal{O}(h^2)$	$\mathcal{O}(h^2)$	$\mathcal{O}(h^0)$	$\mathcal{O}(h^0)$	$\mathcal{O}(h^{-1})$

Table 3: Truncation error order of accuracy of different meshes.

For the randomly perturbed case, a particular random seed was set so that the coefficient results were reproducible for these tests. The first nonzero leading order error coefficient indicates the order of accuracy. This is summarized in table 3. These coefficients have been confirmed to have the same value using the numerical approach (investigated in section IV.B) on the same meshes. It is important to note that for the Green-Gauss scheme as described, the order of accuracy of the truncation error was found to be $\mathcal{O}(h^{-1})$ for randomly perturbed meshes. Experiments reveal that the distinction between the pointwise value at the vertices and the value at the centroids is a source of this behavior. This was done using control volume centroid locations as the reference points for the calculation of moments, gradients, and control volume averages instead of using the vertices as the reference points. The analysis verifies that the truncation error is indeed $\mathcal{O}(h^0)$ instead of $\mathcal{O}(h^{-1})$ when the reference location is changed from the vertex to the centroid. The difference in interpretation leads to subtleties in the error analysis, which warrants further investigation, as examined in section IV.B.

IV.A.1. Jump term coefficient

An interesting result from Nishikawa¹⁰ on the jump term coefficient is that a value of $\alpha = \frac{4}{3}$ will yield fourth order accuracy for a uniform quadrilateral mesh, using Arithmetic Averaging. This possibility is investigated in solving Poisson's equation using the vertex-centered approach on a uniform triangular mesh. We use the computer algebra system for the analytic tests in section IV.A to implement a jump term for each of the dual faces. The discretized flux with jump term, Eq. (12), assumes the form as shown for the Arithmetic Averaging scheme,

$$\nabla\phi = \frac{1}{2}(\nabla\phi^L + \nabla\phi^R) + \frac{\alpha}{|\mathbf{r}_{ij} \cdot \hat{\mathbf{n}}_F|}(\phi^R - \phi^L)\hat{\mathbf{n}}_F. \quad (20)$$

The discrete flux integral for a uniform mesh was calculated analytically as per Eq. (16). Carrying the value of α through as a parameter, the leading order truncation error assumes a simple form for Poisson's equation, as

$$\tau_i = \widetilde{\nabla^2\phi_i} - \overline{\nabla^2\phi_i} = -\frac{3}{16}h^2 \left[\left(\alpha - \frac{4}{3} \right) \frac{\partial^4\phi}{\partial x^4} \Big|_i + 2 \left(\alpha - \frac{4}{3} \right) \frac{\partial^4\phi}{\partial x^2\partial y^2} \Big|_i + \left(\alpha - \frac{4}{3} \right) \frac{\partial^4\phi}{\partial y^4} \Big|_i \right] + \mathcal{O}(h^4). \quad (21)$$

From this expression it is clear that for $\alpha = \frac{4}{3}$, a fourth order accurate truncation error will result, as opposed to a second order accurate truncation error for the uniform mesh without a jump term. However, for irregular meshes, a gain in order of accuracy was not observed by using any value of α due to the lack in symmetry. The truncation error does not assume a form as simple as the one for a uniform triangular mesh. Having the results for the uniform mesh, we will investigate for general meshes if either the truncation error or discretization error is reduced for α near this value. This will be done for different averaging schemes as well, and the results will be compared with the Arithmetic Averaging scheme.

IV.B. Unstructured Mesh - Truncation Error

Even for a slightly perturbed mesh in the regular topology case, the truncation error was found in the analytic tests to be zeroth order accurate using the least-squares scheme. Therefore, for this diffusion scheme on general unstructured meshes, the truncation error is expected to be zeroth order accurate as well. Having obtained results for truncation error for different classes of meshes with regular topology, we proceed to investigate cases of a general triangular unstructured mesh, where the number of first neighbors of a vertex is not always six, and connectivity information needs to be known. We do not preclude the existence of an optimal value or a small range values α that minimizes truncation error in some sense for unstructured meshes. The existence of such an optimal jump term coefficient was similarly observed, having a value of $\frac{2}{3}$ for the cell-centered control volumes in previous work by Jalali and Ollivier-Gooch.⁷ The numerical truncation error was determined by computing the flux integral of the exact solution specific to particular discretization schemes. A simple brute-force search approach was used to find an optimal value of jump term coefficient α that minimizes the L_2 -norm of truncation error. A precise optimization is not pursued here, for the reason that the mesh and solution dependence can alter the optimal value of jump term coefficient slightly for the considered cases. A reference optimal value was determined for the isotropic square mesh with exact solution $u(x, y) = \sin(\pi x) \sin(\pi y)$.

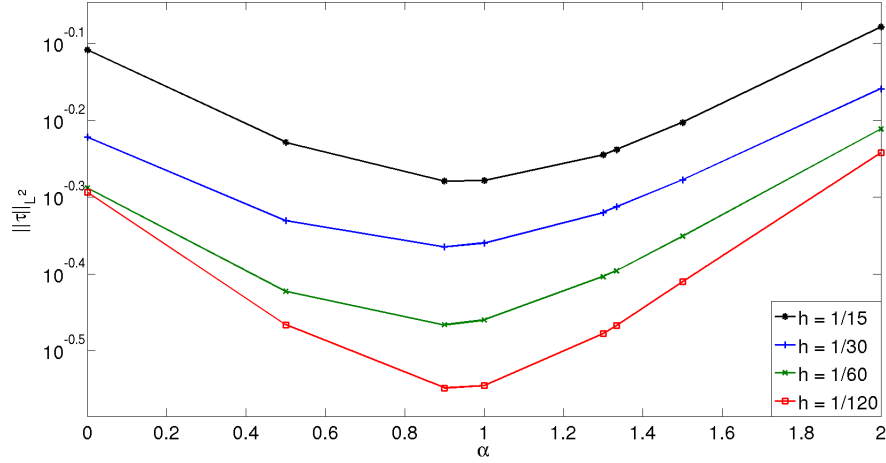


Figure 5: Variation of truncation error with α using Arithmetic Averaging, for four refinements of a square mesh with length scale h , for the exact solution $u(x, y) = \sin(\pi x) \sin(\pi y)$.

The variation of the truncation error with α is shown in figure 5. The optimal value here is taken to be approximately 1.0. This was repeated for the another exact solution $u(x, y) = 20(x - 1) \log(x + 1)y(y - 1)$, shown in figure 6. This particular solution was selected to obtain results for an asymmetric solution but still retaining isotropy. The optimal value of α that minimizes the L_2 -norm of the truncation error deviates substantially from the result obtained for the former exact solution, having a value of approximately 0.4, suggesting a dependence of the truncation error on solutions and derivatives. It is clear that the utility of an optimal jump term coefficient that has a strong dependence on the exact solution is limited at best, since the exact solution for realistic problems is not known *a priori*.

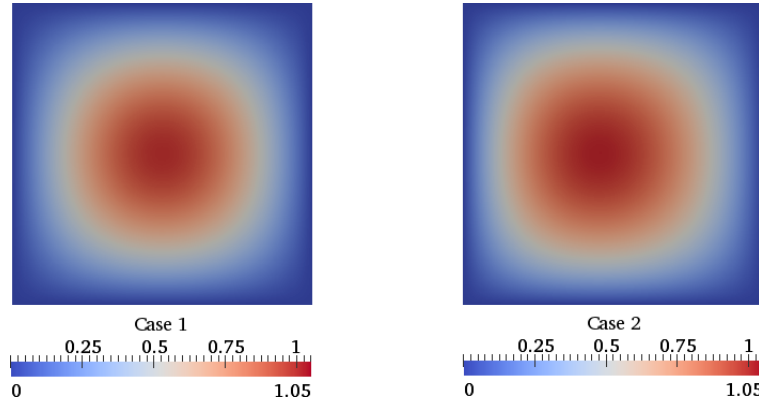


Figure 6: Isotropic exact solutions: Case 1- $u(x, y) = \sin(\pi x) \sin(\pi y)$, Case 2- $u(x, y) = 20(x - 1) \log(x + 1)y(y - 1)$.

An alternate approach is to consider the approach in the analytic case of expressing the truncation error as a series expansion in the solution and solution derivatives. To obtain a single scalar quantity that measures the error, we define a truncation error metric E_i for each control volume as the Euclidean norm of the leading order truncation error coefficients. For schemes having truncation error that is zeroth order, this is expressed as

$$E_i = (\lambda_{xx}^2 + \lambda_{xy}^2 + \lambda_{yy}^2)^{\frac{1}{2}}. \quad (22)$$

A plot of the truncation error metric profile for a square mesh is shown in figure 7. Vertices with at least one neighbor on the boundary were excluded in the analysis to avoid complications due to boundary effects. The error metric was defined in this way to measure the terms that contribute most. Because Eq.

(18) is used to compute the numerical truncation error from the coefficients, a smaller error metric will yield truncation error of smaller magnitude, except in special cases where there is cancellation after substituting in the derivatives of a particular exact solution at the reference points. However, these derivatives of the exact solution are not known in general, so this cannot be exploited. The truncation error metric, as defined, will be solution independent, and is valid provided the solution derivatives in the x and y directions have comparable magnitudes. This error metric has the property that the leading order truncation error is captured in an average sense.

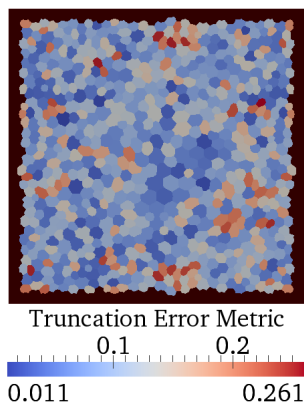


Figure 7: Truncation error metric profile for Arithmetic Averaging, $\alpha = 0$ for a square mesh.

It was verified that adding a jump term in the form of Eq. (12) does not affect the order of accuracy, but does change the error coefficients. The area-weighted average (in the L_1 sense) of the truncation error metric was computed for four refinement levels of an isotropic square mesh. The variation of the area-weighted truncation error metric with jump term coefficient α was investigated, and it was found that there is an optimal jump term coefficient in minimizing this quantity. For the vertex-centered case, the optimal value was found to be approximately 1.0, for the four square meshes considered. The trend of the variation of area-weighted truncation error metric \bar{E} with respect to α using Arithmetic Averaging for one of these meshes is shown in figure 8. Similar values for the other refinement levels were observed.

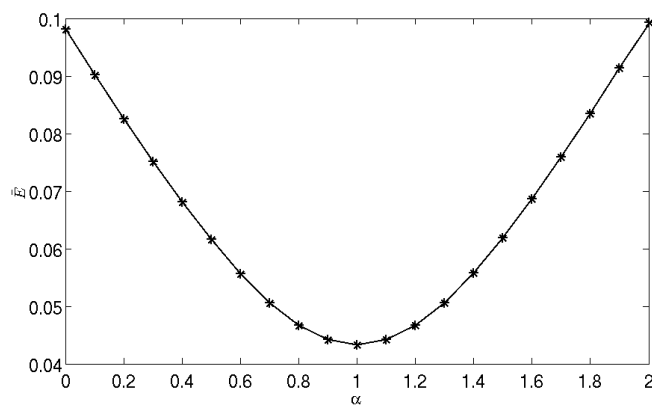


Figure 8: Jump term coefficient effect on truncation error metric for a square mesh.

It is noted that even though the optimal value of α which minimizes the truncation error metric for a given scheme is the same for different refinement levels, there is a variation, albeit small, of this optimal value between different averaging schemes. For the cases tested, all schemes for all isotropic meshes have a variation in the optimal value of α of less than 20%. A comparison of the truncation error metric for the same square mesh is shown in figure 9. These results indicate that the truncation error metric does not depend strongly on the mesh refinement. It is also observed that the Volume-Weighted scheme has a truncation error metric that is approximately an order of magnitude higher than the other schemes.

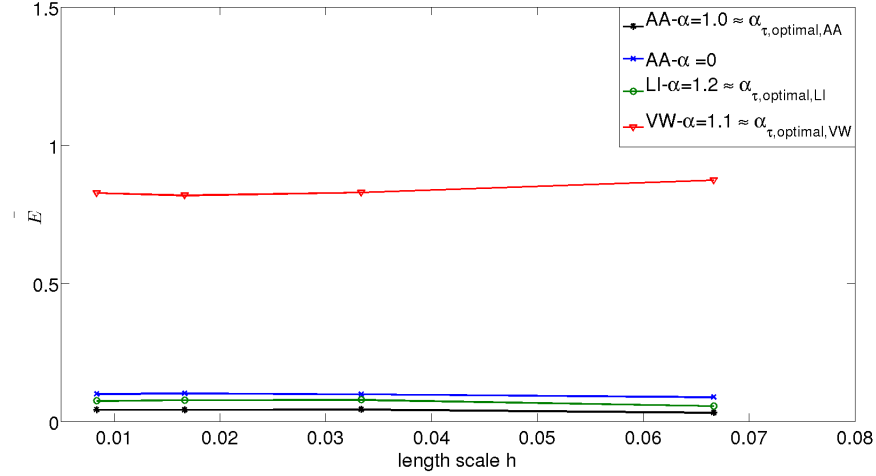


Figure 9: Area averaged truncation error metric for different averaging schemes on four refinements of a square mesh.

IV.C. Unstructured Mesh - Discretization Error

IV.C.1. Isotropic meshes

Motivated by the existence of a particular choice of α that improves truncation error, as discussed, an analogous optimal value that minimizes discretization error is sought. A similar approach was attempted for finding an optimal value of jump term coefficient, α_{optimal} , that in some sense minimizes discretization error. The approach taken is to find the optimal value of α that minimizes the L_2 -norm of the numerical discretization error. The optimization is carried out on four mesh refinement levels of a square for a particular exact solution $u(x, y) = \sin(\pi x) \sin(\pi y)$. The effect of varying α on discretization error for the Arithmetic Averaging scheme is shown in figure 10.

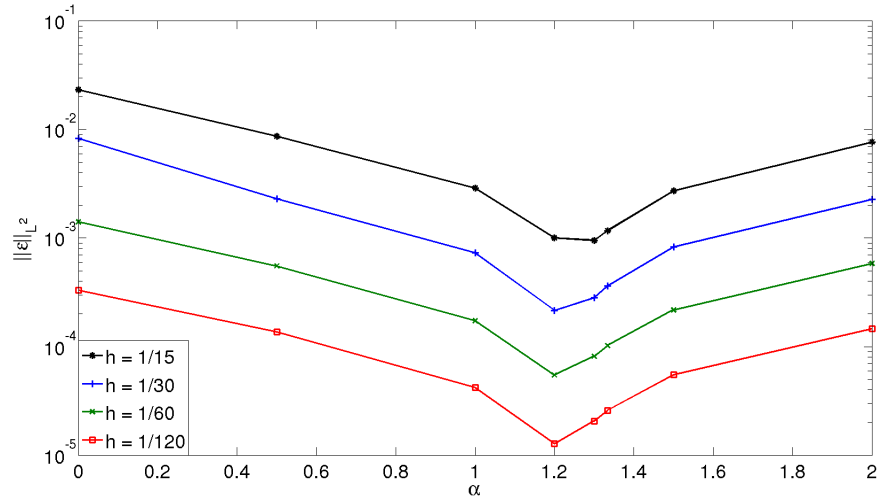


Figure 10: Variation of discretization error with α using Arithmetic Averaging, for four refinements of a square mesh with length scale h , for the exact solution $u(x, y) = \sin(\pi x) \sin(\pi y)$.

Three exact solutions were compared on this sequence of meshes. The value of α that minimizes the discretization error in the L_2 sense was within 20% of 1.2 and was mostly consistent within the four refinement levels. Having the reference optimal value of α that minimizes discretization error for each averaging scheme, a comparison of two different isotropic solutions on the same sequence of meshes is shown in figure 11. One anisotropic solution that has a derivative in the y -direction that is roughly an order of magnitude higher

than that in the x -direction is shown in figure 12. It is found that varying the exact solution does not have a large effect on the optimal value of α for the different averaging schemes.

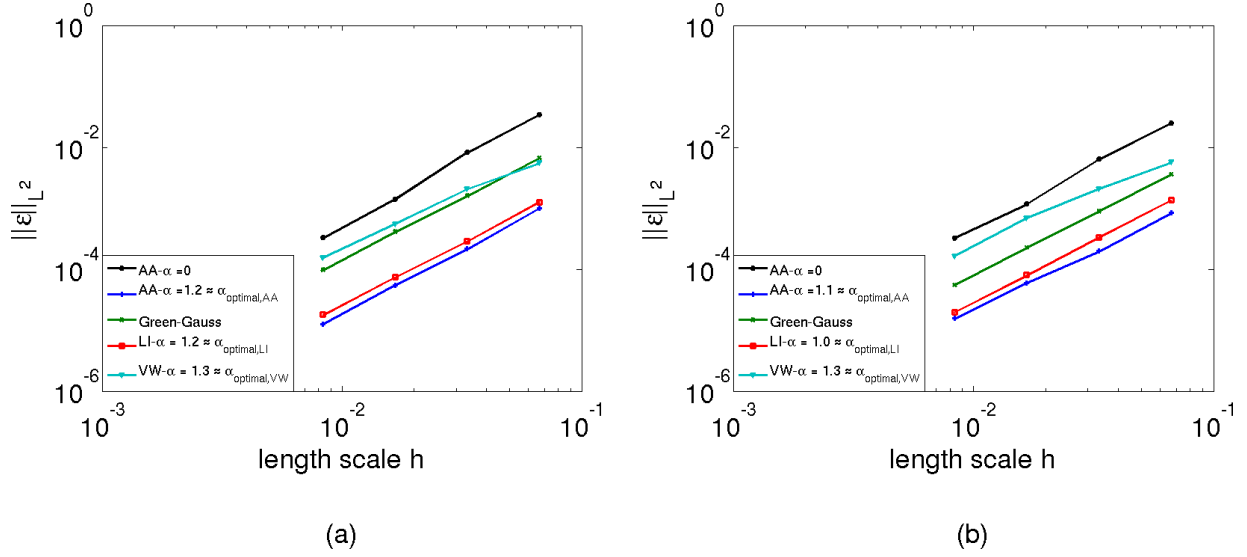


Figure 11: Refinements of the isotropic square mesh $(x, y) \in [0, 1] \times [0, 1]$ for two exact solutions: (a) $u(x, y) = \sin(\pi x) \sin(\pi y)$, and (b) $u(x, y) = 20(x - 1) \log(x + 1)y(y - 1)$

The discretization error of the control volume averages for the Green-Gauss scheme is observed to be first order, when the vertex location is used as the reference location for control volumes. Similar to the discussion above relating to the order of the truncation error, an improved order of accuracy (second order) for discretization error in the control volume averages is observed when the control volume centroid locations are used instead as the reference location, because the control volume average is within second order of the pointwise value at the centroid. The results for the Green-Gauss scheme presented assumes this latter interpretation. It is worthwhile to mention that although it is valid to do the comparison of pointwise error at the vertices for the former interpretation, it may not be simple to generalize to different schemes such as advection-diffusion, where the pointwise error may be second order for some point in the control volume, but not necessarily at the vertex or centroid.

The optimal jump term coefficient was found to be approximately $\alpha_{\text{optimal}} = 1.2$, within a 20% variation in the meshes and solutions tested. Using a value close to optimal value for the jump term coefficient improves the discretization error independent of solution and mesh refinement for isotropic meshes. Additional testing done with different meshes on rectangular domains yielded a similar value for the α_{optimal} , suggesting that for isotropic cases, the optimal jump term coefficient that minimizes discretization error is not strongly dependent on mesh or solution.

IV.C.2. Anisotropic meshes

Some testing was done using the class of anisotropic meshes described by Diskin et al.,⁶ as follows. The mesh is generated on a rectangular domain $\Omega = \{(x, y) | (x, y) \in [0, 1] \times [0, 1]\}$, with $N = (N_x + 1)(N_y + 1)$ vertices. The maximum aspect ratio is denoted by \mathcal{A} . The rows are stretched towards the horizontal center line $y = 0.05$ by a stretching factor β . The y -coordinates can be expressed as

$$y_k = y_{k-1} + \frac{\beta^{k-(N_y/2+1)}}{\mathcal{A} N_x}, \quad y_{\frac{N_y}{2}+1} = 0.05, \quad k = \frac{N_y}{2} + 2, \dots, N_y, N_{y+1}.$$

From here, irregularities are introduced by small random perturbations of interior vertices in the horizontal and vertical directions. The quadrilaterals are then triangulated by selecting one of the two possible diagonals randomly. The maximum aspect ratio \mathcal{A} is chosen to be 100. Four refinement levels are considered, with 9×17 , 17×33 , 33×65 , and 65×129 vertices having stretching factors of 1.340, 1.170, 1.085, and 1.043 respectively. The mesh with 33×65 vertices is shown in figure 13.

One of the exact solutions considered by Diskin et al. for this class of meshes was $u(x, y) = \sin(\pi x + 2\pi y)$. It is noted that this anisotropic mesh is not well suited for the manufactured solution. In the work of Diskin

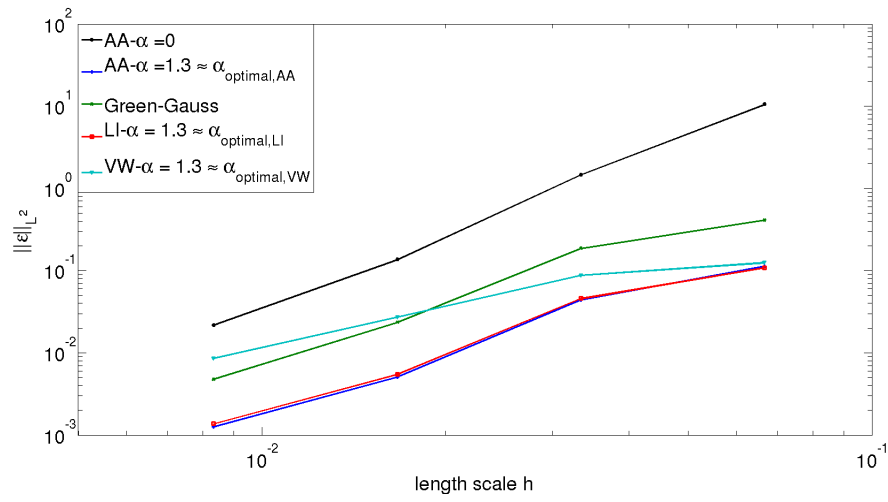


Figure 12: Refinements of the isotropic square mesh $(x, y) \in [0, 1] \times [0, 1]$ for anisotropic solution $u(x, y) = \sin(\pi x) \sin(10\pi y)$

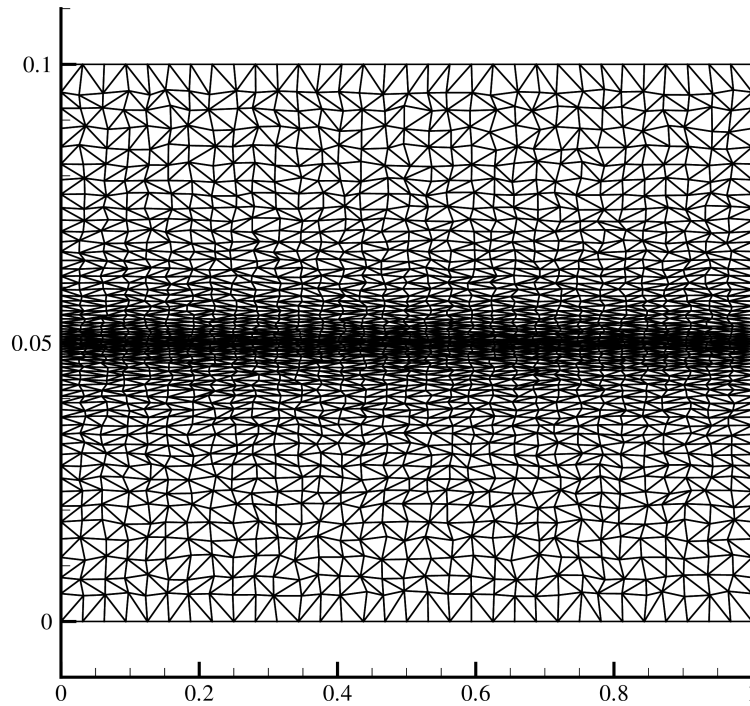


Figure 13: Anisotropic mesh with 33×65 vertices.

et al., this was selected to observe convergence in the worst case. Similarly, we consider the comparison of different diffusion schemes for an exact solution $u(x, y) = \sin(\pi x) \sin(10\pi y)$, which also does not match the mesh anisotropy very well. We would not expect good performance here in terms of accuracy for this reason. In the anisotropic case we use $\alpha = 1.2$, noting that the value that minimizes the discretization error for this particular case deviates substantially from this value. We do not consider optimizing the jump term coefficient for anisotropic grids, for the reason that the mesh anisotropy can vary greatly and the optimal value was observed to depend appreciably on the anisotropy. The accuracy results are plotted in figure 14.

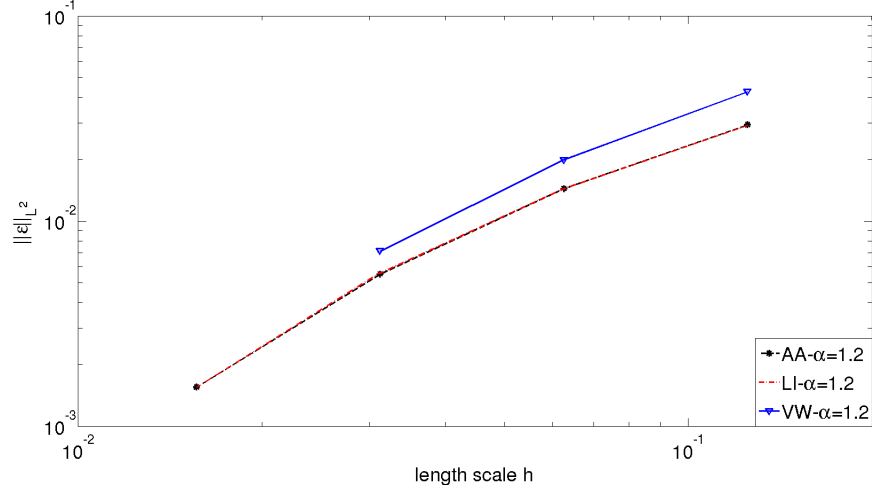


Figure 14: Convergence comparison of schemes to exact solution $u(x, y) = \sin(\pi x) \sin(10\pi y)$ on anisotropic meshes. The Arithmetic Averaging and Linear Interpolation schemes are almost indistinguishable. The Volume-Weighted scheme does not converge on the finest mesh.

It is observed that for fine enough anisotropic meshes of this kind, schemes sometimes fail to converge. The current Green-Gauss implementation does not converge for the two finest meshes. The data from the two coarsest meshes for the Green-Gauss scheme suggests that the asymptotic range has not been reached yet, and so it is not presented here. There is also a missing data point for the finest mesh with Volume-Weighted averaging, where again the scheme suffers from the failure to converge.

A comparison of the discretization error for these schemes suggests that the best schemes for discretization error for the different solutions and meshes tested are Linear Interpolation and Arithmetic Averaging least squares schemes, using the optimal values of α . This is consistent with the observation made in the truncation error analysis, where the Volume-Weighted scheme had a consistently larger truncation error metric than the other two least squares schemes.

IV.D. Jump Term Stability Considerations

For isotropic meshes, it is observed that for any of the cases considered, the solution using Volume-Weighted averaging of the flux does not converge in the absence of a jump term coefficient ($\alpha = 0$). On the anisotropic mesh with the non-matching solution anisotropy the situation is even more severe; none of the averaging schemes (Arithmetic Averaging, Volume-Weighted, Linear Interpolation) converge in the absence of a jump term. However, it is also noted that the presence of a jump term does not necessarily imply convergence of the scheme. With a choice of α large enough, the scheme also becomes unstable. From these observations, it can be seen that there are good choices of α which have a stabilizing effect, but characterizing this possible dependence on scheme, mesh, or solution is deferred for possible future work.

V. Conclusions

Quantification and analysis of truncation error and discretization error were carried out for discretization schemes of diffusion problems. This was performed in the context of a second order vertex-centered finite-volume approach. For the least-squares reconstruction schemes, different averaging methods in which sensible computations of gradients at dual faces between control volumes, each having different physical

interpretations, were analyzed. The addition of a jump term to alleviate the discontinuity across faces was explored. The Green-Gauss scheme, where gradients are computed directly, was also considered.

For simple classes of topologically regular meshes, a truncation error analysis involving the coefficients of the series expansion was carried out using an analytic approach. For geometries with sufficient symmetry, the second order accuracy of truncation error was preserved. This includes the uniform, sheared, and scaled meshes. For the curved, stretched, and perturbed classes of meshes it was found that the truncation error is in general zeroth order accurate. By incorporating the jump term coefficient α , it was found that for the specific uniform mesh, a particular choice of $\frac{4}{3}$ gives a fourth order truncation error, and this motivated the search of optimal jump term coefficients that minimize truncation error and discretization error for the vertex-centered unstructured mesh case.

The analysis for the unstructured mesh case was performed using a numerical approach. For least-squares schemes it was found that an optimal jump term coefficient $\alpha \approx 1.0$ exists in minimizing the truncation error metric, independent of solution, and independent of mesh for isotropic cases. It was found that this does not depend strongly on the type of averaging scheme, as the variation between schemes was observed to be less than 20%. A similar approach was taken to investigate the existence of an optimal jump term coefficient to minimize the L_2 -norm of discretization error. For isotropic meshes, it was found that an optimal value of $\alpha \approx 1.2$ is a good choice independent of solution or mesh. Again, there is some small variation between averaging schemes but the optimal value stays within 20% as well. The proximity of the optimal value of α that minimizes the truncation error metric and discretization error demonstrates some relationship between these notions of error, even though correlations in their magnitudes cannot always be clearly discerned. For the anisotropic meshes considered, a consistent optimal jump term was not found, but a comparison of the accuracy over four levels of refinement is done for the schemes that did converge. The test cases suggest that overall, the Arithmetic Averaging and Linear Interpolation averaging schemes, with the respective optimal jump term coefficient, perform the best in terms of both truncation error and discretization error out of all considered schemes. Here we investigated and quantified some relationships between different discretization schemes, truncation error, and discretization error for diffusion problems. Some insight can be gained in making decisions to possibly reduce numerical error in these models, in addition to motivating applications to more complex physical models.

References

- ¹Mavriplis, D. J., “Grid Resolution Study of a Drag Prediction Workshop Configuration using the NSU3D Unstructured Mesh Solver,” *Proceedings of the Twenty-Third AIAA Applied Aerodynamics Conference*, June 2005, AIAA paper 2005-4729.
- ²Levy, D., Laffin, K. R., Tinoco, E. N., Vassberg, J. C., Mani, M., Rider, B., Rumsey, C. L., Wahls, R. A., Morrison, J. H., Brodersen, O. P., Crippa, S., Mavriplis, D. J., and Murayama, M., “Summary of Data from the Fifth AIAA CFD Drag Prediction Workshop,” *Proceedings of the Fifty-First AIAA Aerospace Sciences Meeting*, 2013.
- ³Roy, C., “Review of Discretization Error Estimators in Scientific Computing,” *Proceedings of the Forty-Eighth AIAA Aerospace Sciences Meeting*, 2010.
- ⁴Banks, J., Hittinger, J., Connors, J., and Woodward, C., “Numerical Error Estimation for Nonlinear Hyperbolic PDEs via Nonlinear Error Transport,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 213-216, March 2012, pp. 1–15.
- ⁵Zhang, X., Trépanier, J.-Y., and Camarero, R., “A Posterior Error Estimation for Finite-Volume Solutions of Hyperbolic Conservation Laws,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 185, 2000, pp. 1–19.
- ⁶Diskin, B., Thomas, J., Nielsen, E., Nishikawa, H., and White, J., “Comparison of node-centered and cell-centered unstructured finite-volume discretizations. Part I: Viscous fluxes,” *Proceedings of the Forty-Seventh AIAA Aerospace Sciences Meeting*, 2009, AIAA paper 2009-597.
- ⁷Jalali, A. and Ollivier-Gooch, C., “Accuracy Assessment of Finite Volume Discretizations of Diffusive Fluxes on Unstructured Meshes,” *Proceedings of the Fiftieth AIAA Aerospace Sciences Meeting*, 2012.
- ⁸Barth, T. J. and Frederickson, P. O., “Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction,” AIAA paper 90-0013, Jan. 1990.
- ⁹Ollivier-Gooch, C. F. and Van Altena, M., “A High-order Accurate Unstructured Mesh Finite-Volume Scheme for the Advection-Diffusion Equation,” *Journal of Computational Physics*, Vol. 181, No. 2, 2002, pp. 729–752.
- ¹⁰Nishikawa, H., “Beyond Interface Gradient: A General Principle for Constructing Diffusion Scheme,” *Proceedings of the Fortieth AIAA Fluid Dynamics Conference and Exhibit*, 2010, AIAA paper 2010-5093.
- ¹¹Barth, T. J., “Aspects of Unstructured Grids and Finite-Volume Solvers for the Euler and Navier-Stokes Equations,” *Lecture Series 1994-05*, von Karman Institute for Fluid Dynamics, Rhode-Saint-Genèse, Belgium, March 1994.
- ¹²Michalak, C., *Efficient high-order accurate unstructured finite-volume algorithms for viscous and inviscid compressible flows*, Ph.D. thesis, The University of British Columbia, Department of Mechanical Engineering, 2009.
- ¹³Manteuffel, T. A. and White, A. B., “The Numerical Solution of Second-Order Boundary Value Problems on Nonuniform Meshes,” *Mathematics of Computation*, Vol. 47, 1986, pp. 511–535.