

Economical Third-Order Methods for Accurate Surface Heating Predictions on Simplex Element Meshes

Kyle B. Thompson*
NASA Langley Research Center,
Hampton, VA 23681, USA

Hiroaki Nishikawa[†] and Emmett Padway[‡]
National Institute of Aerospace,
Hampton, VA 23666, USA

Matthew D. O’Connell[§]
NASA Langley Research Center,
Hampton, VA 23681, USA

A node-centered, edge-based finite-volume discretization of the compressible Navier-Stokes equations is presented with the heat flux reformulated as a first-order system. A dissipation vector is derived for the reformulated system, such that the heat flux can be upgraded to $O(h^3)$ on simplex element meshes in the same fashion as the inviscid fluxes. The method of manufactured solutions is used to demonstrate this uniform order property in heat flux. This new system is shown to produce accurate surface heating predictions for high-speed flow over a hemisphere using an anisotropic simplex element mesh, achieving $O(h^3)$ accuracy at relatively low computational cost compared to similar methods.

I. Introduction

Using computational fluid dynamics (CFD) to simulate surface heat transfer rates has become a common practice, particularly for vehicles operating in the hypersonic flow regime. However, practical second-order accurate finite-volume codes that have experienced difficulty achieving accurate surface heat transfer predictions on arbitrary simplex meshes. Gnoffo and White [1] demonstrated that issues with the inviscid flux function cascade to the surface, corrupting the accuracy of the surface heat flux solution. In this study, a modified form of the Symmetric Total Variation Diminishing (STVD) flux function was employed with moderate effect on improving the stagnation heat transfer predictions on some simplex meshes. Subsequent work by Gnoffo [2] slightly improved the ability to obtain accurate surface heat flux on simplex meshes; however, these improvements were modest at best, with the simplex meshes presented having been generated by subdividing a mixed-element mesh to obtain a simplex mesh.

In practice, CFD practitioners using standard second-order finite-volume CFD solvers are forced to generate structured meshes, or pseudostructured mixed-element meshes to obtain acceptable surface heat transfer results when simulating hypersonic flow. Mccloud [3] details a shock-fitting approach that significantly improves the accuracy of heat flux predictions for blunt body flows, using a process analogous to the shock alignment features available in the LAURA [4] and DPLR [5] structured mesh solvers. Most recently, some progress has been made to further automate unstructured mesh adaptation targeting hypersonic blunt body flows [6, 7]. However, none of these have demonstrated acceptable accuracy in surface heat transfer when a fully tetrahedral mesh is used, and still rely on introducing mixed elements, particularly in boundary layer regions, to improve the surface heat transfer accuracy. The work in this paper demonstrates an improvement to the state of the art by presenting economical third-order accurate finite-volume formulations to accurately predict surface heat transfer on simplex element meshes. These methods are considered economical in the sense that more accurate surface heat flux predictions can be obtained with little to no increase in the total degrees of freedom compared to a conventional vertex-based unstructured second-order finite-volume discretization.

*Research Scientist, Aerothermodynamics Branch, Member AIAA.

[†]Research Fellow, Associate Fellow AIAA.

[‡]Research Scholar, Member AIAA.

[§]Research Scientist, Member AIAA.

II. Governing Equations

A. Navier-Stokes Equations

Consider the steady-state compressible Navier-Stokes equations:

$$\begin{aligned}\nabla \cdot (\rho \mathbf{v}) &= 0 \\ \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} - \boldsymbol{\tau}) + \nabla p &= 0 \\ \nabla \cdot (\rho H \mathbf{v} - \boldsymbol{\tau} \mathbf{v} + \mathbf{q}) &= 0\end{aligned}\tag{1}$$

$$\mathbf{v} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}, \quad \boldsymbol{\tau} = \mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^t - \frac{2}{3} \nabla \cdot \mathbf{v} \mathbf{I} \right), \quad \mathbf{q} = k \nabla T,\tag{2}$$

$$\rho E = \rho \left(e + \frac{1}{2} |\mathbf{v}|^2 \right), \quad \rho H = \rho E + p\tag{3}$$

where $\boldsymbol{\tau}$ and $\mathbf{q}$ are the viscous stress tensor and heat flux vector, respectively. The pressure, p , and internal energy, e , are computed using a calorically perfect gas assumption:

$$p = \rho R T, \quad e = C_v T, \quad C_v = \frac{R}{\gamma - 1}, \quad \gamma = \frac{C_p}{C_v}\tag{4}$$

using a defined specific gas constant, R , and ratio of specific heats, γ . The dynamic viscosity, μ , is computed by Sutherland's Law:

$$\mu = \mu_0 \frac{T_0 + C}{T + C} \left(\frac{T}{T_0} \right)^{3/2},\tag{5}$$

and the thermal conductivity, k , is computed from a specified Prandtl number, Pr :

$$k = \frac{\gamma}{\gamma - 1} \frac{\mu}{Pr} R\tag{6}$$

where μ_0 is the reference viscosity defined at the reference temperature T_0 , C is the Sutherland constant prescribed for the fluid.

B. Hyperbolic Navier-Stokes - HNS14

To improve the order of accuracy of the stress tensor, $\boldsymbol{\tau}$, and heat flux, $\mathbf{q}$, Eq. 1 can be recast into a first-order system of hyperbolic equations [8], referred to as HNS14:

$$\nabla \cdot (\rho \mathbf{v}) = 0\tag{7}$$

$$\nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} - \boldsymbol{\tau}) + \nabla p = 0\tag{8}$$

$$\nabla \cdot (\rho H \mathbf{v} - \boldsymbol{\tau} \mathbf{v} + \mathbf{q}) = 0\tag{9}$$

$$\frac{t_v}{\mu_v} \partial_t (\boldsymbol{\tau}) - \nabla \mathbf{v} = -\frac{\boldsymbol{\tau}}{\mu_v}\tag{10}$$

$$\frac{t_h}{\mu_h} \partial_t (\mathbf{q}) - \nabla \left(\frac{a^2}{\gamma(\gamma - 1)} \right) = -\frac{\mathbf{q}}{\mu_h}\tag{11}$$

where μ_v and μ_h are scaled viscosities,

$$\mu_v = \frac{4}{3} \mu, \quad \mu_h = \frac{\gamma \mu}{Pr}\tag{12}$$

a is the speed of sound,

$$a = \sqrt{\gamma R T}\tag{13}$$

and t_v and t_h are relaxation times associated with the viscous stress and heat flux

$$t_v = \frac{\rho}{\mu_v} L_d^2, \quad t_h = \frac{\rho}{\mu_h} L_d^2, \quad L_d = \frac{1}{2\pi}\tag{14}$$

Note that pseudotime derivatives have been added to the additional viscous stress and heat flux equations, which allow relaxation of τ and q and make these equations hyperbolic in pseudotime. There are many choices to discretize this system of equations, and this formulation of the Navier-Stokes equations and its extensions [9, 10] have been demonstrated to produce more accurate surface shear stress and heat flux predictions in finite-volume discretizations than a traditional discretization of Eq. 1. In this paper, the focus is on deriving a discretization that effectively predicts the heat flux at surface boundaries for simplex element meshes, with a minimum set of equations. With this goal in mind, we formulate a new system that only includes the heat flux variables.

C. First-order Heat Flux Reformulation

If only heat flux accuracy improvements are desired, an economical alternative to the HNS14 system can be formulated by dropping the viscous stress equations, i.e., Eq. 10:

$$\nabla \cdot (\rho \mathbf{v}) = 0 \quad (15)$$

$$\nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} - \boldsymbol{\tau}) + \nabla p = 0 \quad (16)$$

$$\nabla \cdot (\rho H \mathbf{v} - \boldsymbol{\tau} \mathbf{v} + \mathbf{q}) = 0 \quad (17)$$

$$\frac{t_h}{\mu_h} \partial_t (\mathbf{q}) - \nabla \cdot \left(\frac{a^2}{\gamma(\gamma - 1)} \right) = -\frac{\mathbf{q}}{\mu_h} \quad (18)$$

This modification is useful for hypersonic flow simulations, where accurate heat transfer and mitigation of the additional number of equations to be solved are both top priorities. Because the original HNS14 system was derived using a calorically perfect assumption, it is not directly usable for reacting flow simulations. To formulate a similar system that is extendable to reacting flow simulations, the μ_h viscosity can be exchanged with thermal conductivity, k , to reformulate the heat flux equations as:

$$\frac{t_h}{k} \partial_t (\mathbf{q}) - \nabla T = -\frac{\mathbf{q}}{k}. \quad (19)$$

However, this was found to result in a very numerically stiff system of equations when discretized using the finite-volume scheme. To improve the numerical properties of the discretization, an even simpler system is derived, where the temperature gradient can be solved for instead of the heat flux variables, $\mathbf{q}$, similar to the HNS20G scheme by Li et al. [11]:

$$\begin{aligned} \nabla \cdot (\rho \mathbf{v}) &= 0 \\ \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} - \boldsymbol{\tau}) + \nabla p &= 0 \\ \nabla \cdot (\rho H \mathbf{v} - \boldsymbol{\tau} \mathbf{v} + \mathbf{q}) &= 0 \\ t_h \partial_t (\mathbf{g}_T) - \nabla T &= -\mathbf{g}_T \end{aligned} \quad (20)$$

where,

$$\mathbf{g}_T = \nabla T, \quad \mathbf{q} = -k \mathbf{g}_T, \quad t_h = \frac{\rho}{k} L_d^2 \quad (21)$$

The work presented in this paper focuses on obtaining higher-order accuracy in the heat flux solution by using Eq. 20 to introduce three additional equation to be solved. As in Ref. [11], a frozen thermal conductivity assumption, $\frac{\partial k}{\partial \mathbf{Q}} = 0$, is used to simplify the derivation of numerical dissipation required for upwinding. This derivation is carried out in Appendix A.

III. Edge-Based Finite-Volume Discretization

Using a node-centered edge-based finite-volume method, we discretize Eq. 20 as a system of coupled equations advanced in pseudotime in the form:

$$\mathbf{P}_j^{-1} \frac{\partial \mathbf{Q}_j}{\partial t} V_j + \sum_{k \in \{k_j\}} (\boldsymbol{\Phi}_{jk} A_{jk}) = \mathbf{S}_j V_j \quad (22)$$

where $\mathbf{Q}_j$ is the conserved variable state vector at node j , V_j is the dual control volume around node j , $\boldsymbol{\Phi}_{jk}$ is the numerical flux directed through the dual control volume face area, A_{jk} , at node j associated with edge k with $\{k \in \{k_j\}\}$

being all edge neighbors of node j . To achieve third-order accuracy, we also upgrade the source term quadrature as we will discuss later, which results in a completely edge-based loop for the spatial terms in the discretization,

$$\mathbf{P}_j^{-1} \frac{\partial \mathbf{Q}_j}{\partial t} V_j + \sum_{k \in \{k_j\}} [(\Phi_{jk} + \mathbf{S}_{jk}) A_{jk}] = 0 \quad (23)$$

where $\mathbf{S}_{jk}$ is the source term contribution to node j associated with edge k .

A. Numerical Fluxes

As is consistent with previous work on hyperbolic forms of the Navier-Stokes equations [8, 9, 11, 12], the numerical Eq. 20 fluxes are split into distinct systems:

$$\Phi = \Phi^i + \Phi^v + \Phi^g \quad (24)$$

where the inviscid fluxes, Φ^i , and heat fluxes, Φ^g , are upwinded and the remaining viscous shear fluxes, Φ^v , are computed using the alpha-damping scheme [13]. The upwind fluxes can be expressed as a combination of consistent and dissipation terms:

$$\Phi^i = \frac{1}{2} [(\mathbf{F}_n^i)_L + (\mathbf{F}_n^i)_R] - \frac{1}{2} |\mathbf{A}_n^i| \Delta \mathbf{Q} \quad (25)$$

$$\Phi^g = \frac{1}{2} [(\mathbf{F}_n^g)_L + (\mathbf{F}_n^g)_R] - \frac{1}{2} \mathbf{P}^{-1} |\mathbf{P} \mathbf{A}_n^g| \Delta \mathbf{Q} \quad (26)$$

$$\mathbf{Q} = \begin{pmatrix} \rho \\ \rho \mathbf{v} \\ \rho E \\ \mathbf{g}_T \end{pmatrix}, \quad \mathbf{F}_n^i = \begin{pmatrix} \rho \mathbf{v}^t \\ \rho \mathbf{v} \otimes \mathbf{v} + p \mathbf{I} \\ \mathbf{v}^t (\rho E + p) - (\tau \mathbf{v})^t \\ \mathbf{0} \end{pmatrix} \mathbf{n}, \quad \mathbf{F}_n^g = \begin{pmatrix} 0 \\ \mathbf{0} \\ \mathbf{q} \\ T \mathbf{I} \end{pmatrix} \mathbf{n} \quad (27)$$

$$\mathbf{P}^{-1} = \text{diag} (1, \dots, t_h, t_h, t_h) \quad (28)$$

where the subscripts L and R denote the left and right sides of the edge midpoint, $\Delta Q = Q_R - Q_L$, $\mathbf{n}$ is the unit normal vector to the control volume face, and $\mathbf{A}_n^i$ and $\mathbf{A}_n^g$ are the normal flux Jacobians,

$$\mathbf{A}_n^i = \frac{\partial \mathbf{F}_n^i}{\partial \mathbf{Q}}, \quad \mathbf{A}_n^g = \frac{\partial \mathbf{F}_n^g}{\partial \mathbf{Q}}. \quad (29)$$

In this work, $\mathbf{A}_n^i$ is computed using the Roe-averaged state and $\mathbf{A}_n^g$ is computed using the arithmetic average. The dissipation term for the heat flux, Φ^g is derived in Appendix A as,

$$\mathbf{P}^{-1} |\mathbf{P} \mathbf{A}_n^g| \Delta \mathbf{Q} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{k}{L_d} \Delta T \\ \frac{L_d}{k} (\Delta \mathbf{q} \cdot \mathbf{n}) \mathbf{n} \end{pmatrix}. \quad (30)$$

The remaining viscous flux, Φ^v , is computed as

$$\Phi^v = \frac{1}{2} [(\mathbf{F}_n^v)_L + (\mathbf{F}_n^v)_R] \quad (31)$$

$$\mathbf{F}_n^v = \begin{pmatrix} 0 \\ -\tau \\ -(\tau \mathbf{v})^t \\ \mathbf{0} \end{pmatrix} \mathbf{n} \quad (32)$$

$$\tau = \mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^t - \frac{2}{3} \nabla \cdot \mathbf{v} \mathbf{I} \right), \quad \nabla \mathbf{v} = \frac{1}{2} [\nabla \tilde{\mathbf{v}}_L + \nabla \tilde{\mathbf{v}}_R] - \frac{\alpha}{L_r} \Delta \mathbf{v} \quad (33)$$

where $\nabla \tilde{\mathbf{v}}_{L/R}$ are the gradients computed by a linear (or quadratic) least-squares method at the left and right nodes of the edge, α is a tunable parameter, and $L_r = |\mathbf{e}_{jk} \cdot \mathbf{n}_{jk}|$ is a geometric skewness measurement used to increase damping when the edge vector, $\mathbf{e}_{jk} = \mathbf{x}_k - \mathbf{x}_j$, is misaligned with the unit normal of the dual face, $\mathbf{n}_{jk}$.

B. Higher-order Reconstruction

To better resolve smooth flow features, the left and right edge midpoint states can be extrapolated from the endpoint nodes. This reconstruction at the edge midpoint is done using the following

$$\mathbf{U} = (\rho, \mathbf{v}, T, \mathbf{g}_T) \quad (34)$$

with the left and right states computed from two endpoint nodes, j and k , respectively,

$$\mathbf{U}_L = \mathbf{U}_j + \frac{1}{2} \nabla \mathbf{U}_j, \quad \mathbf{U}_R = \mathbf{U}_k + \frac{1}{2} \nabla \mathbf{U}_k \quad (35)$$

and evaluating the fluxes with these reconstructed states yields second-order accuracy when the gradients are computed from linear least-squares (LSQ) fit. Katz and Sankaran [14] demonstrated that third-order accuracy in the convective fluxes can be achieved on simplex meshes through linear flux extrapolation, provided that the flux gradients are computed by an algorithm that is exact for a quadratic function. The linear flux extrapolation can be performed efficiently using the chain rule:

$$\mathbf{F}_L = (\mathbf{F}_n)_j + \frac{1}{2} \left(\frac{\partial \mathbf{F}_n}{\partial \mathbf{U}} \right)_j \nabla \mathbf{U}_j \cdot \mathbf{e}_{jk}, \quad \mathbf{F}_R = (\mathbf{F}_n)_k - \frac{1}{2} \left(\frac{\partial \mathbf{F}_n}{\partial \mathbf{U}} \right)_k \nabla \mathbf{U}_k \cdot \mathbf{e}_{jk} \quad (36)$$

so that we only need to compute the solution gradients by a quadratic method. These gradients are also used in the linear reconstruction for the left and right states. Then, the solution jump will vanish for quadratic functions as shown in Ref. [14]. In this work, the gradients are computed using a quadratic least-squares (QLSQ) technique, where the coefficients for the gradient at nodes are computed and stored and the Hessian coefficients are discarded, using edge-connected neighbor nodes and their neighbors.

C. Robust Flux Functions and Reconstruction Limiting

In the presence of strong shock waves, the reconstruction procedure must be limited to avoid spurious oscillations. Typically, a slope limiter is employed on the gradients to enforce strict monotonicity, such as the multidimensional limiting procedure (MLP) [15] of Park and Kim. To reduce the requirement of strict monotonicity to prevent unrealizable reconstructed states, the robust flux formulation derived by Nishikawa [16] is used. This is a natural extension of the linear flux extrapolation, whereby the dissipation matrix terms, $\mathbf{A}_n^i$ and $\mathbf{A}_n^g$, are evaluated using the solution at nodes, rather than the reconstructed face state; therefore, only the jump terms use reconstructed values, which have no realizability restrictions provided that the jump terms are purely functions of the solution, $\mathbf{U}$. When gradients are constructed with linear least-squares (LLSQ), the consistent part of flux is evaluated as

$$\mathbf{F}_L = (\mathbf{F}_n)_j, \quad \mathbf{F}_R = (\mathbf{F}_n)_k, \quad (37)$$

which maintains second-order accuracy on an edge-based discretization where the face centroid is collocated with the edge midpoint, and avoids evaluating thermodynamic state equations with potentially unphysical conditions, such as negative density or negative temperature. When gradients are evaluated with QLSQ, the consistent part of the flux is evaluated using Eq. 36, which improves the order of accuracy without requiring evaluations using the reconstructed states.

A heuristic pressure limiter derived by Gnoffo and White [1]

$$\hat{\phi}_j = \min \left[1, \max \left(0, \frac{\phi_{max} - \phi_{jk}}{\phi_{max} - \phi_{min}} \right) \right] \quad (38)$$

is employed to suppress oscillations at shock waves without resorting to using a conventional monotonicity-preserving slope limiter, where ϕ_{jk} is the ratio of maximum to minimum pressure at all edge-neighbors of a vertex j , and $\phi_{min/max}$ are tunable parameters. For this work, $\phi_{max} = 3$ and $\phi_{min} = 2$ were found to be reasonable for values. The pressure

limiter is computed at each node and is premultiplied with the nodal gradient used in all state and flux extrapolations in the place of a convention slope limiter would be employed in a stencil-based total variable bounding (TVB) scheme. Note that here Eq. 38 is employed directly to limit the gradients used for reconstruction, whereas in Ref. [1], it is used for additional limiting on a top of a conventional limiter. Using these techniques was found to improve iterative convergence, so that the nonlinear solver is able to routinely achieve machine-zero flux balance for steady-state flow simulations.

D. Third-order Source Term Quadrature

As shown in Eq. 22, the source terms can be integrated with a point-quadrature formula for second-order accuracy. For third-order accuracy, an economical formula has been derived by Nishikawa and Liu [17] that maintains accuracy and only requires the source term gradient at the node associated with the dual control volume (not those at neighbor nodes):

$$\int_{V_j} \mathbf{S} dV = \frac{1}{60} \sum (13\mathbf{S}_j + 3\partial\mathbf{S}_{jk} - 3\mathbf{S}_k) (\mathbf{e}_{jk} \cdot \mathbf{n}_{jk}) \quad (39)$$

$$\mathbf{S} = \mathbf{g}_T, \quad \partial\mathbf{S}_{jk} = (\nabla\mathbf{g}_T)_j \cdot \mathbf{e}_{jk} . \quad (40)$$

Note that the source term, $\mathbf{S}_j$, does not require any additional LSQ gradient calculation, because it can reuse the same quadratic LSQ gradients used in the flux reconstruction.

E. Boundary Flux Quadrature

The fluxes on the domain boundaries must be closed using a sufficiently accurate flux quadrature scheme. If a suitable quadrature scheme is not used, the overall accuracy of the solution will degrade from design order. For second-order accuracy, the flux quadrature formulas in Ref. [12] are used. On boundary elements attached to tetrahedra, the second-order accurate general boundary quadrature formula is

$$\left(\frac{6}{8}\mathbf{F}_{jB} + \frac{1}{8}\mathbf{F}_2 + \frac{1}{8}\mathbf{F}_3 \right) \cdot \mathbf{n}_B \quad (41)$$

where n_B is the directed area of the triangle and $\mathbf{F}_{jB}$, $\mathbf{F}_2$, and $\mathbf{F}_3$ are the fluxes evaluated at the corners of the triangle using the boundary condition state and the state at nodes j , 2, and 3, respectively.

For third-order accuracy, the third-order preserving flux quadrature scheme [18] is used. On boundary elements attached to tetrahedra, the third-order accuracy preserving general boundary quadrature formula is

$$\left(\frac{1}{2}\mathbf{F}_{jB} + \frac{1}{4}\mathbf{F}_{j2} + \frac{1}{4}\mathbf{F}_{j3} \right) \cdot \mathbf{n}_B \quad (42)$$

where $\mathbf{n}_B$ is the directed area of the triangle, $\mathbf{F}_{jB}$ is the flux evaluated using the boundary condition state and the state at node j of the triangle, and $\mathbf{F}_{j2}$, and $\mathbf{F}_{j3}$ are fluxes evaluated using interior higher order reconstruction scheme on the $j2$ and $j3$ triangle edges, respectively.

Farfield, no-slip surface, and outflow boundary conditions are employed in this work. The farfield boundary condition state, U_{farfield} , is imposed as a Dirichlet boundary condition

$$U_{\text{farfield}} = \begin{pmatrix} \rho_{\infty} \\ \mathbf{v}_{\infty} \\ T_{\infty} \\ \mathbf{g}_{T_{\text{interior}}} \end{pmatrix} \quad (43)$$

where ρ_{∞} is the freestream density, $\mathbf{v}_{\infty}$ is the freestream velocity, T_{∞} is the freestream temperature, and $\mathbf{g}_{T_{\text{interior}}}$ are the temperature gradient variables extrapolated from the interior. The outflow boundary condition state is extrapolated from the interior, $U_{\text{outflow}} = U_{\text{interior}}$. The no-slip surface boundary condition, U_{surface} is strongly imposed as an isothermal wall condition

$$U_{\text{surface}} = \begin{pmatrix} \rho_{\text{interior}} \\ \mathbf{0} \\ T_{\text{wall}} \\ \mathbf{g}_{T_{\text{interior}}} \end{pmatrix} \quad (44)$$

where T_{wall} is the user-specified wall temperature. After all fluxes and source terms are integrated, the momentum and energy equations at nodes associated with no-slip surface boundary conditions are replaced with algebraic equations that strongly impose a zero velocity condition and wall temperature condition.

IV. HyperSolve CFD Solver

The aforementioned discretizations have been implemented in the HyperSolve CFD solver, which is being developed at the NASA Langley Research Center. HyperSolve is a fluid solver dynamically loadable plugin in the T-infinity ecosystem [19], and uses a Jacobian-Free Newton Krylov (JFNK) method to iteratively converge the solution to steady state. This JFNK method is implemented through the Enigma library [20], which provides a simplified interface to the PETSc [21] library. In this work, the additive-Schwarz-ILU0 preconditioner with level-1 domain overlap is used to right-precondition the GMRES Krylov linear solver in PETSc. As described in Ref. [20], both pseudotransient continuation (PTC) and a backtracking line search algorithm are used to globalize the inexact Newton's method.

V. Results

The new first-order heat flux system is compared with a traditional discretization based on the alpha-damping scheme. In all cases, $\alpha = 4/3$ is used in the alpha-damping scheme, which has been demonstrated to provide good iterative convergence and accuracy [13]. To improve robustness and accuracy in high Reynolds number flows, the diffusion length scale, L_d , is replaced with:

$$L_h = \frac{L_d}{\sqrt{Re_h}}, \quad Re_h = \frac{\rho_\infty |\mathbf{v}_\infty| L_d}{\mu_h} \quad (45)$$

as suggested by Yi and Nishikawa [22]. An accuracy verification study is performed with these parameters, as well as a three-dimensional hemisphere case that demonstrates the effectiveness of the reformulated first-order heat flux system as compared to the alpha-damping scheme.

A. Accuracy Verification

The method of manufactured solutions is used to verify the order of accuracy of the implemented numerical scheme. A simple exponential function is used as the manufactured solution

$$\begin{aligned} \rho &= 0.11 e^x + 0.12 e^y + 0.13 e^z + 1.23 \\ u &= 10 e^x + 20 e^y + 30 e^z + 100 \\ v &= 10 e^x + 20 e^y + 30 e^z + 200 \\ w &= 10 e^x + 20 e^y + 30 e^z + 300 \\ T &= 10 e^x + 20 e^y + 30 e^z + 350 \end{aligned} \quad (46)$$

with each solution variable nondimensionalized by the freestream reference state. The L_1 norm of the discretization error is measured on a series of tetrahedral meshes on the cubic domain, with the manufactured solution weakly imposed as a Dirichlet condition at all boundary nodes. The accuracy of the gradients is expected to be one order less than the solution for the traditional alpha damping scheme. Meaning the use of QLSQ gradients to achieve $O(h^3)$ accuracy in temperature results in $O(h^2)$ accuracy in the temperature gradients. The first-order reformulation of the heat fluxes, however, should result in uniform $O(h^3)$ accuracy for both temperature and temperature gradients. This is confirmed by Figure 1, which shows that the convergence of the L_1 norm of discretization error matches the expected design order of both the alpha-damping scheme and first-order heat flux system when QLSQ gradients are used in conjunction with the required boundary and source quadratures and appropriate flux extrapolation.

B. Hemisphere

To demonstrate the robustness and effectiveness of the first-order reformulation in high-speed flow, we consider a hemisphere in freestream conditions of $\rho_\infty = 0.01 \text{ kg/m}^3$, $U_\infty = 1000 \text{ m/s}$, $T_\infty = 300 \text{ K}$, $\gamma = 1.4$, and $Pr = 0.72$. The mesh used is a fully-tetrahedral mesh generated from the *Sketch-to-Solution* (S2S) workflow [6]. The adaptation procedure was initialized from hemisphere domain geometry definition, and iteratively adapted by controlling interpolation error

of the temperature solution field with using the *refine** library. The LAURA [4] solver is used to provide a benchmark solution, because it is considered to be a standard tool in aerothermodynamic analysis of external hypersonic flows. Figure 2a shows a comparison of the LAURA mesh (left half) and the final surface mesh produced from the S2S adaptation procedure (right half), and Figure 2b shows the surface pressure contours on the adapted tetrahedral mesh along with a centerline slice of the tetrahedral mesh. Figure 3 compares the surface heat flux contours obtained from LAURA and the various methods described here on the adapted tetrahedral mesh. The P1 continuous Galerkin technique computes the gradients of the viscous fluxes (including the heat flux) elementwise via gradients computed by the Green-Gauss method, and is included as reference solution representing a state-of-the-art technique in conventional second-order finite-volumes solvers. Both the P1 continuous Galerkin method and alpha-damping edge-based scheme (with Linear LSQ gradients) show poor agreement with the LAURA result, although the alpha-damping result in Figure 3b is qualitatively more useful than the P1 continuous Galerkin result in Figure 3a. Both the alpha-damping edge-based scheme (with Quadratic LSQ gradients) and the reformulated heat flux system significantly improve the smoothness of the surface heat transfer contours and are in better agreement with the LAURA solution. Figure 4 shows a comparison of the surface heat flux values predicted by HyperSolve along a $Y=0$ centerline slice, as well as error bars showing $\pm 5\%$ of the LAURA solution. The largest variation in heat flux is observed near the stagnation point, with both third-order scheme predictions being within 3% of the predicted stagnation heating value. The reformulated first-order system provides the smoothest heat flux predictions, albeit all results are in relatively close agreement. The P1 Galerkin scheme results are not shown, because they would be off the plot.

VI. Conclusions and Future Work

An economical scheme involving recasting heat flux in the Navier-Stokes equations as a first-order system is presented. Accuracy verification has been done using the method of manufactured solution to demonstrate the expected $O(h^3)$ design order of accuracy is achieved uniformly for both the temperature and heat flux solution variables in the reformulated system. This new system demonstrates an improvement over a traditional finite-volume viscous discretization in its ability to accurately predict surface heat flux in high-speed flow on an anisotropic tetrahedral mesh.

Future work will focus on quantifying and improving the efficiency of the third-order edge-based discretizations described here. The ability to achieve third-order accurate surface heat flux predictions with only three additional degrees of freedom per mesh vertex is a unique advantage of the reformulated first order-system described here. However, this still does increase the memory requirement of the Jacobian used to precondition the linear solver, which may prohibit its use in reacting flow simulations with a large number of species. Future work will investigate approximations that can be employed in the Jacobian to minimize the additional memory requirement and computational expense, as well as quantify the accuracy improvement relative to the increased computational expense of the reformulated first-order system compared to the alpha-damping scheme.

Finally, future work will also investigate the use of the improved gradient accuracy from these schemes when forming the adaptation metric needed for anisotropic simplex mesh adaptation methods. Traditional second-order finite-volume schemes typically can only provide $O(1)$ Hessians needed for adaptation metrics, whereas the third-order edge-based methods described here can provide $O(h)$ Hessians and even $O(h^2)$ Hessians for constructing the adaptation metric. There is potential to substantially decrease the overall time to solution of sufficient accuracy by employing more advanced anisotropic mesh adaptation techniques that target higher truncation error terms than is typically employed for the standard second-order finite-volume schemes being widely used today.

A. Eigenstructure and Dissipation Vector Derivation

The eigenstructure for the fully coupled reformulated first-order heat flux system in Eq. 20 has not been found; however, deriving the dissipation term for the heat flux equations alone makes the analysis straightforward. First, we

*refine is developed by NASA and is available via <https://github.com/NASA/refine> under the Apache License, Version 2.0, last accessed November, 2022.

define conserved state vector, $\mathbf{Q}$, normal flux, $\mathbf{F}_n$, and local preconditioning matrix, $\mathbf{P}$ as

$$\mathbf{Q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \\ q_x \\ q_y \\ q_z \end{pmatrix}, \quad \mathbf{F}_n = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \mathbf{q} \\ T n_x \\ T n_y \\ T n_z \end{pmatrix} \quad (47)$$

$$\mathbf{P} = \text{diag} (1, 1, 1, 1, 1, t_h/k, t_h/k, t_h/k) \quad (48)$$

and then deriving the eigenstructure of the preconditioned three-dimensional heat flux normal Jacobian,

$$\mathbf{P}\mathbf{A}_n = \mathbf{P} \frac{\partial \mathbf{F}_n}{\partial \mathbf{Q}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & n_x & n_y & n_z \\ a n_x & b n_x & c n_x & d n_x & e n_x & 0 & 0 & 0 \\ a n_y & b n_y & c n_y & d n_y & e n_y & 0 & 0 & 0 \\ a n_z & b n_z & c n_z & d n_z & e n_z & 0 & 0 & 0 \end{pmatrix}, \quad (49)$$

where a, b, c, d , and e are the temperature linearizations for a perfect gas

$$\begin{aligned} a &= \frac{k}{t_h} \frac{\partial T}{\partial \rho} = -a_h^2 (E - |\mathbf{v}|^2) \\ b &= \frac{k}{t_h} \frac{\partial T}{\partial \rho u} = -a_h^2 u \\ c &= \frac{k}{t_h} \frac{\partial T}{\partial \rho v} = -a_h^2 v \\ d &= \frac{k}{t_h} \frac{\partial T}{\partial \rho w} = -a_h^2 w \\ e &= \frac{k}{t_h} \frac{\partial T}{\partial \rho E} = a_h^2 \end{aligned} \quad (50)$$

The eigenvalues and right eigenvectors of Eq. 49 are

$$\lambda_1 = -\sqrt{e} \quad \lambda_2 = \sqrt{e} \quad \lambda_{3,4,5,6,7,8} = 0 \quad (51)$$

$$\mathbf{R} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{a}{2e} & \frac{a}{2e} & -\frac{a}{e} & -\frac{b}{e} & -\frac{c}{e} & -\frac{d}{e} & 0 & 0 \\ -\frac{a n_x}{2\sqrt{e}} & \frac{a n_x}{2\sqrt{e}} & 0 & 0 & 0 & 0 & \frac{n_x^2}{n_x^2 + n_z^2} & 0 \\ -\frac{a n_y}{2\sqrt{e}} & \frac{a n_y}{2\sqrt{e}} & 0 & 0 & 0 & 0 & 0 & \frac{n_z^2}{n_y^2 + n_z^2} \\ -\frac{a n_z}{2\sqrt{e}} & \frac{a n_z}{2\sqrt{e}} & 0 & 0 & 0 & 0 & -\frac{n_x n_z}{n_x^2 + n_z^2} & -\frac{n_y n_z}{n_y^2 + n_z^2} \end{pmatrix}. \quad (52)$$

Solving for the left eigenvectors

$$\mathbf{L} = \begin{pmatrix} 1 & \frac{b}{a} & \frac{c}{a} & \frac{d}{a} & \frac{e}{a} & -\frac{\sqrt{e}}{a}n_x & -\frac{\sqrt{e}}{a}n_y & -\frac{\sqrt{e}}{a}n_z \\ 1 & \frac{b}{a} & \frac{c}{a} & \frac{d}{a} & \frac{e}{a} & \frac{\sqrt{e}}{a}n_x & \frac{\sqrt{e}}{a}n_y & \frac{\sqrt{e}}{a}n_z \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -\frac{n_x}{n_z} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\frac{n_y}{n_z} \end{pmatrix}, \quad (53)$$

which satisfies

$$\mathbf{L}\mathbf{R} = \mathbf{I}. \quad (54)$$

The dissipation matrix is obtained as

$$|\mathbf{P}\mathbf{A}_n| = \mathbf{R}|\Lambda|\mathbf{L} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{a}{\sqrt{e}} & \frac{b}{\sqrt{e}} & \frac{c}{\sqrt{e}} & \frac{d}{\sqrt{e}} & \sqrt{e} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{e}n_x^2 & \sqrt{e}n_xn_y & \sqrt{e}n_xn_z \\ 0 & 0 & 0 & 0 & 0 & \sqrt{e}n_xn_y & \sqrt{e}n_y^2 & \sqrt{e}n_y n_z \\ 0 & 0 & 0 & 0 & 0 & \sqrt{e}n_xn_z & \sqrt{e}n_y n_z & \sqrt{e}n_z^2 \end{pmatrix}, \quad (55)$$

where $a_h = \sqrt{\frac{k}{C_v t_h \rho}}$ is the thermal diffusion wavespeed, we obtain the final result for the dissipation matrix

$$|\mathbf{P}\mathbf{A}_n| = \begin{pmatrix} \mathbf{0} & \mathbf{0}^t & \mathbf{0} & \mathbf{0} \\ -a_h(E - |\mathbf{v}|) & -a_h\mathbf{v} & a_h & 0 \\ \mathbf{0} & \mathbf{0}^t & \mathbf{0} & a_h\mathbf{n} \otimes \mathbf{n} \end{pmatrix}. \quad (56)$$

And the dissipation vector

$$\mathbf{P}^{-1}|\mathbf{P}\mathbf{A}_n|\Delta\mathbf{Q} = \mathbf{P}^{-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ a_h(\Delta\rho E - (E - |\mathbf{v}|)\Delta\rho - \mathbf{u} \cdot \Delta\rho\mathbf{u}) \\ a_h(\Delta\mathbf{q} \cdot \mathbf{n})\mathbf{n} \end{pmatrix}. \quad (57)$$

Equation 57 can be further simplified by using a change of variables,

$$\mathbf{P}^{-1}|\mathbf{P}\mathbf{A}_n|\Delta\mathbf{Q} = \mathbf{P}^{-1}|\mathbf{P}\mathbf{A}_n|\frac{\partial\mathbf{Q}}{\partial\mathbf{U}}\Delta\mathbf{U} \quad (58)$$

$$\mathbf{U} = \begin{pmatrix} \rho \\ \mathbf{v} \\ T \\ \mathbf{q} \end{pmatrix}, \quad \frac{\partial\mathbf{Q}}{\partial\mathbf{U}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \mathbf{v} & \rho\mathbf{I} & 0 & \mathbf{0}^t \\ E & \rho\mathbf{v}^t & \rho C_v & 0 \\ 0 & \mathbf{0}^t & 0 & \mathbf{I} \end{pmatrix}, \quad (59)$$

which simplifies the heat flux dissipation vector to

$$\mathbf{P}^{-1}|\mathbf{P}\mathbf{A}_n|\frac{\partial \mathbf{Q}}{\partial \mathbf{U}}\Delta \mathbf{U} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{k}{L_d}\Delta T \\ \frac{L_d}{k}(\Delta \mathbf{q} \cdot \mathbf{n})\mathbf{n} \end{pmatrix}. \quad (60)$$

Finally, note that the dissipation vector for the heat fluxes and variable set in Eq. 20 is identical to Eq. 60 when a frozen thermal conductivity assumption is used, $\frac{\partial k}{\partial Q} = 0$. This is because the heat normal flux Jacobians of Eq. 15 and Eq. 20 are related by symmetric scaling,

$$S_k = \text{diag}(1, 1, 1, 1, 1, k, k, k) \quad (61)$$

meaning the Jacobian matrices only differ in their eigenvectors. Applying the symmetric scaling to Eq. 56 yields the same final the dissipation vector result as Eq. 60.

Acknowledgments

This work was supported by the NASA Langley Science Directorate and the NASA Science Mission Directorate. The first author would like to thank the Entry Systems Modeling Project within the NASA Game Changing Development Program, Space Technology Mission Directorate for their funding and support of this research. The second and third authors gratefully acknowledge support by the U.S. Army Research Office under the contract/grant number W911NF-19-1-0429. Also, the authors would like to thank Dr. Kyle Anderson for useful discussion regarding the hyperbolic Navier-Stokes system.

References

- [1] Gnoffo, P., and White, J. A., “Computational Aerothermodynamic Simulation Issues on Unstructured Grids,” AIAA Paper 2004–2371, 2004.
- [2] Gnoffo, P. A., “Simulation of stagnation region heating in hypersonic flow on tetrahedral grids,” AIAA Paper 2007–3960, 2007.
- [3] McCloud, P. L., “Best Practices for Unstructured Grid Shock Fitting,” AIAA Paper 2017–1149, 2017.
- [4] Thompson, K. B., Hollis, B. R., Johnston, C. O., Kleb, B., Lessard, V. R., and Mazaheri, A., “LAURA Users Manual: 5.6,” NASA TM-2020-220566, Langley Research Center, 2020.
- [5] Hash, D., Olejniczak, J., Wright, M., Prabhu, D., Pulsonetti, M., Hollis, B., Gnoffo, P., Barnhardt, M., Nompelis, I., and Candler, G., “FIRE II Calculations for Hypersonic Nonequilibrium Aerothermodynamics Code Verification: DPLR, LAURA, and US3D,” AIAA Paper 2007-605, 2007.
- [6] Kleb, W., Park, M., Wood, W., Bibb, K., Thompson, K., and Gomez, R., “Sketch-to-Solution: An Exploration of Viscous CFD with Automatic Grids,” AIAA Paper 2019–2948, 2019.
- [7] Nastac, G., Tramel, R. W., and Nielsen, E. J., “Improved Heat Transfer Prediction for High-Speed Flows over Blunt Bodies using Adaptive Mixed-Element Unstructured Grids,” AIAA Paper 2022–0111, 2022.
- [8] Nishikawa, H., “New-Generation Hyperbolic Navier-Stokes Schemes: $O(1/h)$ Speed-Up and Accurate Viscous/Heat Fluxes,” AIAA Paper 2011–3043, 2011.
- [9] Nakashima, Y., Watanabe, N., and Nishikawa, H., “Hyperbolic Navier-Stokes Solver for Three-Dimensional Flows,” AIAA Paper 2016–1101, 2016.
- [10] Liu, Y., and Nishikawa, H., “Third-Order Inviscid and Second-Order Hyperbolic Navier-Stokes Schemes for Three-Dimensional Inviscid and Viscous Flows,” AIAA Paper 2016–3969, 2016.
- [11] Li, L., Lou, J., Nishikawa, H., and Luo, H., “Reconstructed Discontinuous Galerkin Methods for Compressible Flows Based on a New Hyperbolic Navier-Stokes system,” *Journal of Computational Physics*, Vol. 427, 2020, p. 110058.

- [12] Nishikawa, H., “Beyond Interface Gradient: A General Principle for Constructing Diffusion Schemes,” AIAA Paper 2010–5093, 2010.
- [13] Nishikawa, H., Nakashima, Y., and Watanabe, N., “Effects of High-Frequency Damping on Iterative Convergence of Implicit Viscous Solver,” *Journal of Computational Physics*, Vol. 348, 2017.
- [14] Katz, A., and Sankaran, V., “Mesh Quality Effects on the Accuracy of CFD Solutions on Unstructured Meshes,” *Journal of Computational Physics*, Vol. 230, 2011, pp. 7670–7686.
- [15] Park, J. S., and Kim, C., “Multi-dimensional limiting process for finite volume methods on unstructured grids,” *Computers and Fluids*, Vol. 65, 2012, pp. 8–24.
- [16] Nishikawa, H., “Robust Numerical Fluxes for Unrealizable States,” *Journal of Computational Physics*, 2020, p. 109244. doi:10.1016/j.jcp.2020.109244.
- [17] Nishikawa, H., and Liu, Y., “Accuracy-Preserving Source Term Quadrature for Third-Order Edge-Based Discretization,” *Journal of Computational Physics*, Vol. 344, 2017.
- [18] Nishikawa, H., “Accuracy-Preserving Boundary Flux Quadrature for Finite-Volume Discretization on Unstructured Grids,” *Journal of Computational Physics*, Vol. 281, 2014, pp. 518–555.
- [19] O’Connell, M. D., Druyor, C. T., Thompson, K. B., Jacobson, K. E., Anderson, W. K., Nielsen, E. J., Carlson, J.-R., Park, M. A., Jones, W. T., Biedron, R. T., Kleb, B., and Zhang, X., “T-infinity: The Dependency Inversion Principle for Rapid and Sustainable Multidisciplinary Software Development,” AIAA Paper 2018–3856, 2018.
- [20] Thompson, K. B., and O’Connell, M. D., “Streamlined Convergence Acceleration for CFD Codes,” AIAA Paper 2019-3709, 2019.
- [21] Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., May, D. A., McInnes, L. C., Mills, R. T., Munson, T., Rupp, K., Sanan, P., Smith, B. F., Zampini, S., Zhang, H., and Zhang, H., “PETSc Users Manual,” Tech. Rep. ANL-95/11 - Revision 3.10, Argonne National Laboratory, 2018. URL <http://www.mcs.anl.gov/petsc>.
- [22] Nishikawa, H., and Liu, Y., “Hyperbolic Navier-Stokes Method for High-Reynolds-Number Boundary Layer Flows,” AIAA Paper 2017–0081, 2017.

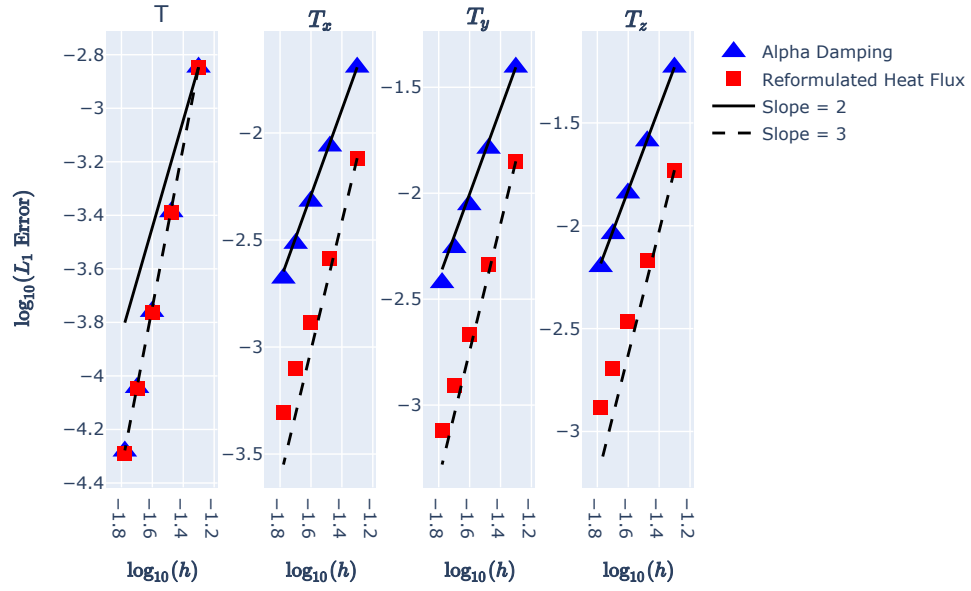
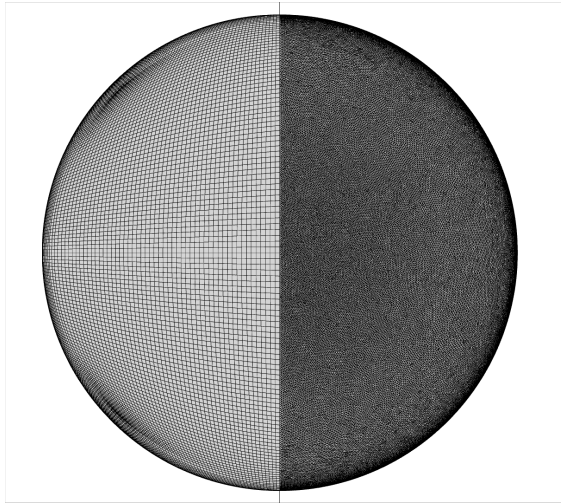
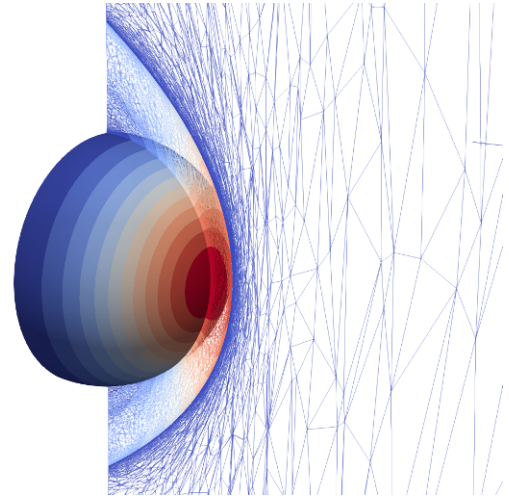


Fig. 1 Method of manufactured solutions discretization error for temperature and temperature gradients.

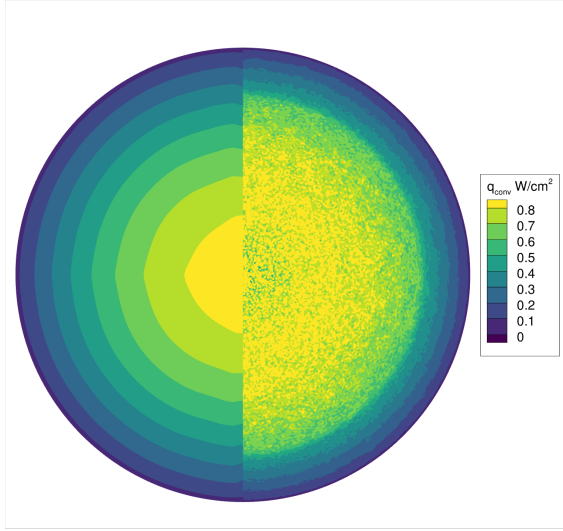


(a) Surface mesh.

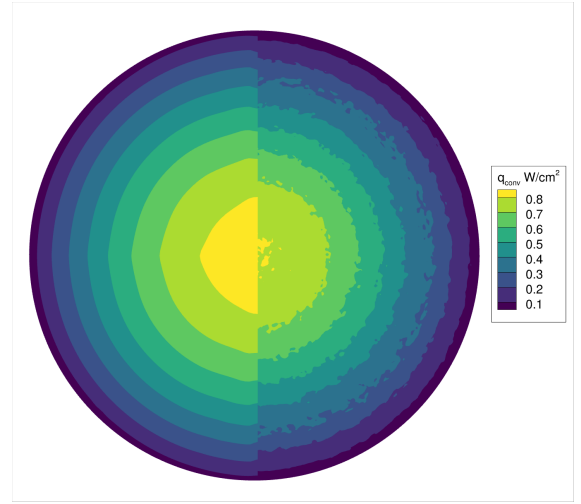


(b) HyperSolve pressure contours.

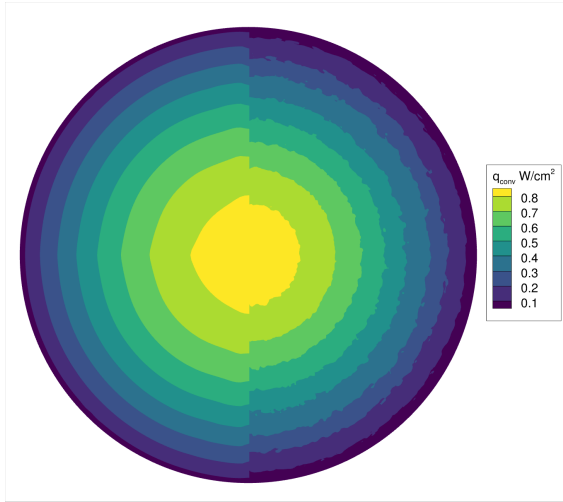
Fig. 2 LAURA multiblock structured mesh vs HyperSolve adapted tetrahedral mesh comparison.



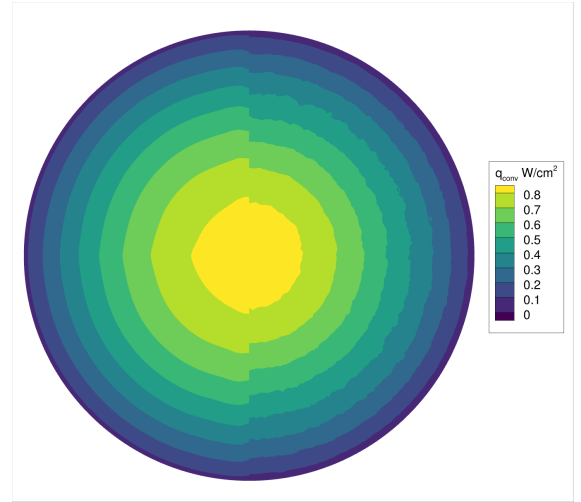
(a) P1 Continuous Galerkin



(b) Alpha-Damping w/Linear LSQ



(c) Alpha-Damping w/Quadratic LSQ



(d) Reformulated Heat Flux

Fig. 3 Hemisphere surface heat flux comparison between (left) LAURA and (right) HyperSolve.

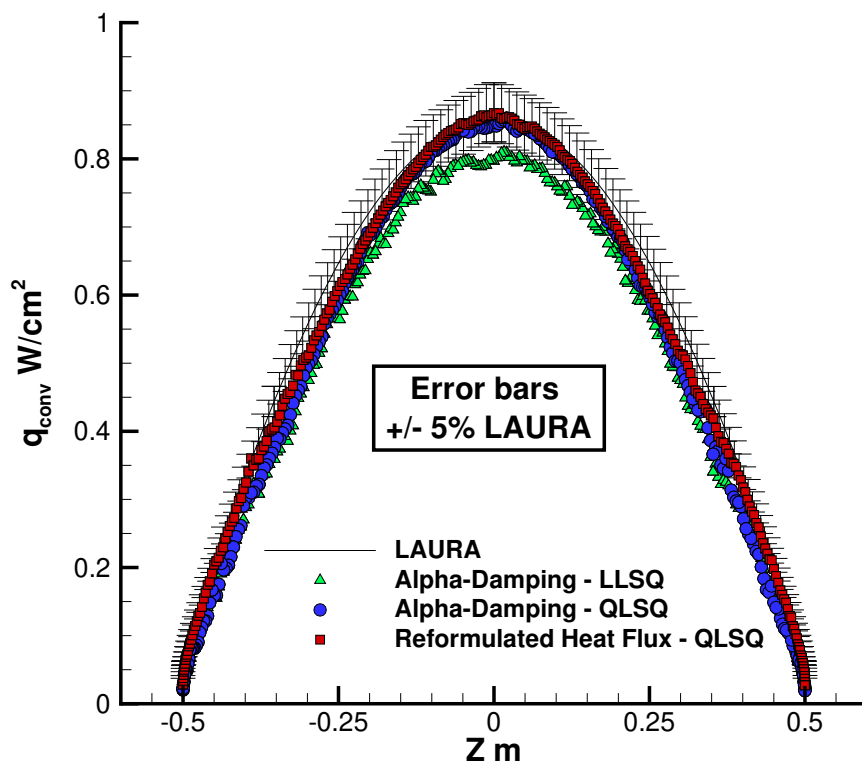


Fig. 4 Hemisphere surface heat flux at $Y=0$ plane.