Development and Verification of the Charring Ablating Thermal Protection Implicit System Solver

Adam J. Amar,[*] Nathan D. Calvert,[*] and Benjamin S. Kirk[*]

NASA Lyndon B. Johnson Space Center
2101 NASA Parkway
Houston, TX, 77058

The development and verification of the Charring Ablating Thermal Protection Implicit System Solver is presented. This work concentrates on the derivation and verification of the stationary grid terms in the equations that govern three-dimensional heat and mass transfer for charring thermal protection systems including pyrolysis gas flow through the porous char layer. The governing equations are discretized according to the Galerkin finite element method with first and second order implicit time integrators. The governing equations are fully coupled and are solved in parallel via Newton's method, while the fully implicit linear system is solved with the Generalized Minimal Residual method. Verification results from exact solutions and the Method of Manufactured Solutions are presented to show spatial and temporal orders of accuracy as well as nonlinear convergence rates.

Nomenclature

α	thermal diffusivity $(\mathrm{m^2/sec})$ or absorptivity
$\alpha_t, \beta_t, \gamma_t$	temporal finite difference weights $(\mathrm{sec^{-1}})$
β	extent of reaction
$\boldsymbol{Q}$	volumetric flow rate $(\mathrm{m^3/sec})$
$\boldsymbol{U}$	independent variable vector
$\boldsymbol{v}$	velocity $(\mathrm{m/sec})$
$\boldsymbol{v}'$	superficial velocity $(\mathrm{m/sec})$
Δh	characteristic mesh length scale (m)
Δt	time step (sec)
$\dot{\boldsymbol{s}}$	surface recession rate $(\mathrm{m/sec})$
$\dot{\omega}$	mass source $(\mathrm{kg/m^3 \cdot sec})$
$\dot{m}$	mass flux $(\mathrm{kg/m^2 \cdot sec})$
$\dot{Q}$	energy source $(\mathrm{W/m^3})$
$\dot{q}$	heat flux $(\mathrm{W/m^2})$
ϵ	emissivity or penalty factor
Γ	volume fraction in virgin material
$\hat{\boldsymbol{n}}$	unit normal vector
$\kappa, \widetilde{\boldsymbol{\kappa}}$	permeability $(\mathrm{m^2})$
$\mathcal{R}$	residual
μ	dynamic viscosity $(\mathrm{Pa \cdot sec})$
ϕ	porosity
ψ	basis function
ρ	density $(\mathrm{kg/m^3})$
σ	Stefan-Boltzmann constant $(\mathrm{W/m^2 \cdot K^4})$
θ	non-dimensional temperature or manufactured solution amplitude
A	area $(\mathrm{m^2})$
A_t, B_t, C_t, D_t	manufactured solution temporal coefficients $(\mathrm{sec^{-1}})$

[*] Applied Aeroscience and CFD Branch, Mail Code EG3, Member AIAA.

A_x, B_y, C_z	manufactured solution spatial coefficients (m^{-1})
C_H	heat transfer Stanton number
C_p	specific heat $(\mathrm{J}/\mathrm{kg}{\cdot}\mathrm{K})$
DE	discretization error
E	activation energy (J/kg) or error
e	internal energy (J/kg)
e_o	total internal energy (J/kg)
H	heat transfer coefficient $(\mathrm{W}/\mathrm{m}^2{\cdot}\mathrm{K})$
h	enthalpy (J/kg)
h^o	enthalpy of formation (J/kg)
h_o	total enthalpy (J/kg)
$J \times W$	Jacobian of the transformation matrix times the integration weight
$k, \widetilde{\boldsymbol{k}}$	pre-exponential factor (sec^{-1}) or thermal conductivity $(\mathrm{W}/\mathrm{m}{\cdot}\mathrm{K})$
L	domain length scale (m)
m	reaction order
N	number of nodes
N_e	number of elements
nc	number of components
P	pressure (Pa)
p	temporal order of accuracy
Q	generic quantity of interest
q	spatial order of accuracy
R	gas constant $(\mathrm{J}/\mathrm{kg}{\cdot}\mathrm{K})$
r	perturbation factor
T	temperature (K)
t	time (sec)
u	velocity component $(\mathrm{m}/\mathrm{sec})$
v	test function

Subscripts and Superscripts

∞	denotes far-field state
ν	iteration index
c	quantity of the char
e	boundary layer edge or element quantity
g	quantity of the gas
h	denotes finite dimensional approximation
m	quantity of the mesh
n	time index
o	denotes total, initial, or reference condition
QP	denotes a value at a quadrature point
r	denotes recovery state
s	quantity of the solid or surface
v	quantity of the virgin
w	quantity of gases adjacent to the wall

I. Introduction

Reliable predictions of the performance of thermal protection systems (TPS) is critical in the design and analysis of atmospheric entry vehicles. For decades, design engineers have relied on numerical heat transfer solutions for the highest fidelity thermal response predictions. For vehicle components exposed to high heat fluxes, greater than $40\,\mathrm{W}/\mathrm{cm}^2$, usually an ablative thermal protection system is required, and they generally tend to be more thermally and chemically complex than reusable TPS materials. These complexities require additional physics to be accounted for in the computational models.

In general, there are two classes of ablators: those that only undergo surface ablation and those that decompose, char, or pyrolyze internally in addition to ablating at the surface, which is depicted in Figure 1. Before entering an atmosphere, a charring ablator TPS material initially starts out as the virgin composite material. The virgin composite is typically a fibrous material injected with resin that binds the material and adds mechanical strength. As the vehicle enters, it is exposed to a heat flux which leads to thermal penetration in-depth. As the material heats up, resin and fiber constituents can pyrolyze, which is typically an endothermic process, resulting in voids and gas generation. The pyrolysis gases are driven through the porous char layer by internal pressure gradients, and the majority of the gas is expelled from the surface into the chemically reacting boundary layer. As the surface temperature rises, boundary layer gases start to react with the exposed char and pyrolysis gas at the surface. The gas/surface chemical reactions result in mass removal from the surface and the ablation products are carried away with the fluid stream. This mass removal causes the surface to recede, and the ablative TPS thins throughout entry.

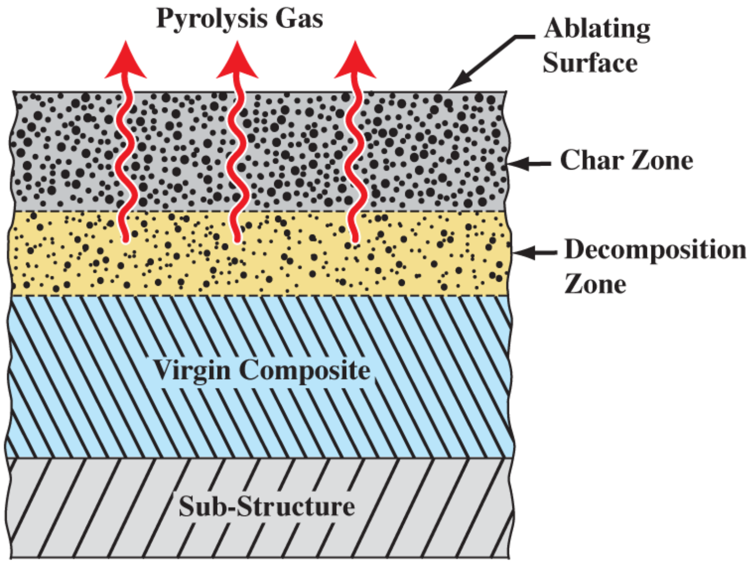


Figure 1. Charring ablator schematic.

There have been many attempts to model the complicated thermochemical processes involved in the charring ablator problem for the past fifty years. The most notable contributions to this technology area were by the Aerotherm Corporation in the late 1960s. They presented the development of the charring material, surface thermochemistry, and reacting boundary layer models used in CMA, ACE, and BLIMP respectively.[1–3] The CMA theory manual written by Moyer and Rindal is of particular interest to this study since it outlines the model for the one-dimensional energy and mass transfer problem within the ablator. Moyer and Rindal proposed an in-depth material model that is still being used today. The model implemented in CMA assumes that the virgin and char thermal properties are known, and the thermal properties in the decomposing region is interpolated between the two. The decomposition of virgin into char and gas is modeled using Arrhenius rate equations. The pyrolysis gases resulting from decompostion of the virgin material have no residence time in the char layer and are instantaneously injected into the boundary layer as soon as they are produced. The pyrolysis gases do not react with the char layer as they percolate to the surface nor do they deposit residue (coke) in the pore space.

Numerous researchers and code developers have adopted the material model in CMA since its introduction and have made incremental improvements including improved mesh motion schemes over the translating/node-dropping scheme in CMA, improved discretization methods and numerical algorithms, extension to multiple dimensions, addition of gas phase continuity equation and porous flow equations, and improved physical models such as thermochemical non-equilibrium in-depth. del Valle, Pike, and April developed a model in which they solved the equilibrium and non-equilibrium porous gas flow equations to determine the pressure and gas composition in-depth.[4,5] Chen and Milos developed the FIAT code, which is one dimensional and has the same material model as CMA, but it is fully implicit.[6] FIAT has been used in numerous TPS design and analysis activities since its development which most recently includes Stardust, Mars Science Laboratory, and Orion.[7–9] Blackwell, Hogan, and Cochran introduced the use of the control volume finite element method (CVFEM) and linear elastic mesh motion to solve one and two dimensional ablation problems.[10–12] Ahn and Suzuki et. al. developed the SCMA and SCMA2 codes that solve the one and two dimensional charring ablation problems including pyrolysis gas flow within the char layer.[13,14] Amar, Blackwell, and Edwards used the CVFEM to solve the coupled energy and gas continuity equations and presented verification studies which proved correct implementation of many of the boundary conditions and terms in the governing equations.[15–17] Recently, Dec and Braun presented codes that solve one and three-dimensional ablation problems including pyrolysis gas flow. Dec and Braun used the Galerkin finite element method (FEM) for spatial discretization and use linear elastic mesh motion.[18,19]

One recurring shortcoming of most of the ablation code development work over the years has been the lack of code verification studies to ensure proper implementation of the governing equations, boundary conditions, and numerical methods employed in the software. While various definitions for "code verification and validation" can be found in literature, the definition for code verification adopted in this work is given by Roache:[20]

> The code author defines precisely what continuum partial differential equations and continuum boundary conditions are being solved, and convincingly demonstrates that they are solved correctly, i.e. usually with some order of accuracy, and always consistently, so that as some measure of discretization (e.g. mesh increments) $\Delta \rightarrow 0$, the code produces a solution to the continuum equations; this is Verification. Whether or not those equations and that solution bear any relation to a physical problem of interest to the code user is the subject of Validation.

In other words, verification is the strictly mathematical exercise of proving that the model equations and numerical methods have been correctly implemented in the software, and validation provides evidence that the correct equations have been chosen to solve the physical problem of interest. While many code developers bypass verification exercises and attempt to validate their codes through comparisons with data, verification is a necessary step that must precede code validation because one can not know if the correct equations are being solved (validation) without first knowing that the equations they intend to solve are correctly implemented (verification). Additionally, code-to-code comparisons and/or extensive use of a code are not part of the rigorous verification process. However, these activities do have merit in that they help build confidence, improve code usability, and demonstrate robustness and reliability. While verification studies should be performed for every term in the governing equations, all boundary conditions, and all modeling options, there is some practical limit to how rigorously a developer can verify a code given that most software of this nature is developed in an environment where sooner than later practical problems need to be solved for engineering design and analysis.

In this study, the development and verification of the Charring Ablating Thermal Protection Implicit System Solver is presented. The code solves the two and three dimensional charring ablator, general porous flow, and heat transfer problems in serial or parallel. The code uses the first and second order implicit time integrators and the Galerkin FEM spatial discretization on unstructured hybrid meshes. Supported element types include quadrilaterals and triangles in two dimensions and tetrahedrons, hexahedrons, prisms, and pyramids in three dimensions. Furthermore, the code is capable of solving problems using adaptive mesh refinement on all element types in both two and three dimensions. The governing equations are fully coupled and solved fully implicitly with Newton's method (with exact Jacobian terms), and the Generalized Minimal Residual method (GMRES) is used to solve the linear system.[21] The exact Jacobian terms are calculated through the use of the complex-step (or complex perturbation) method rather than being analytically derived. This enables optimal convergence of the nonlinear solver as opposed to degraded convergence rates from approximate Jacobians. As far as the authors are aware, the complex-step method is a novel approach to calculating Jacobians for numerical heat transfer or CFD. The method dramatically reduces development effort and the potential for bugs resulting from derivation and coding errors.

This study concentrates on the verification of the energy and porous flow equations in both two and three dimensions, several of their boundary conditions, and the nonlinear system solver. Comparisons with analytical solutions are used when available but the Method of Manufactured Solutions (MMS) is also used to verify the nonlinear heat and mass transfer equations.[20,22] This study only deals with the non-decomposing and stationary mesh terms. While not all of the physical models used in this study are necessarily relevant for a broad range of ablation and porous flow problems, it should be realized that the terms in the governing equations verified here will be present in all charring ablation problems. Consequently, these are the necessary first steps in having a fully, or at least adequately, verified code.

II. Mathematical Model

II.A. Governing Equations

The equations that govern the solid/gas system of the porous charring ablator include energy and mass conservation equations for the solid as well as the Navier-Stokes equations as applied to all of the gaseous species considered. In the general case, it is possible that the pyrolysis gases react with the remaining solid, or deposit residue (coke) on the solid, but these phenomena are neglected. Under the assumptions that the pyrolysis gas is in thermochemical equilibrium and the solid and gas are in thermal equilibrium, then the solid and gas energy equations for a moving mesh reduce to a mixture energy equation given by

$$\frac{\partial(\rho e_o)}{\partial t} = \boldsymbol{\nabla} \cdot \left(\widetilde{\boldsymbol{k}} \boldsymbol{\nabla} T\right) - \boldsymbol{\nabla} \cdot \left(\phi \rho_g h_{o_g} \boldsymbol{v}_g\right) + \boldsymbol{\nabla} \cdot \left(\rho h_o \boldsymbol{v}_m\right) + \dot{Q} \tag{1}$$

The velocity of the pyrolysis gases is governed by a porous flow law such as Darcy's law, which omits the necessity to solve the momentum equations. Since ablators in general can be anisotropic materials, the thermal conductivity is a second order tensor. The solid mass conservation equation is

$$\frac{\partial \rho_s}{\partial t} = \dot{\omega}_s + \boldsymbol{\nabla} \cdot \left(\rho_s \boldsymbol{v}_m\right) \tag{2}$$

If it is assumed that all solid decomposition results in pyrolysis gas generation, $\dot{\omega}_g = -\dot{\omega}_s$, and the gases occupy all of the pore space and are free to flow through it, then the gas mass conservation equation is given by

$$\frac{\partial(\phi \rho_g)}{\partial t} = \dot{\omega}_g - \boldsymbol{\nabla} \cdot \left(\phi \rho_g \boldsymbol{v}_g\right) + \boldsymbol{\nabla} \cdot \left(\phi \rho_g \boldsymbol{v}_m\right) \tag{3}$$

In this study, not every term in the governing equations will be verified. Looking only at the terms that are verified in this work, the equations essentially describe the energy and mass transfer in a non-ablating material (no mesh motion) assuming no in-depth decomposition, no gas flow contributions in the energy equation, and no energy contribution in the gas flow equation. Consequently, the verified equations reduce to

$$\text{Verified Energy Equation: } \frac{\partial(\rho e_o)}{\partial t} = \boldsymbol{\nabla} \cdot \left(\widetilde{\boldsymbol{k}} \boldsymbol{\nabla} T\right) + \dot{Q} \tag{4}$$

and

$$\text{Verified Gas Continuity Equation: } \frac{\partial(\phi \rho_g)}{\partial t} = \dot{\omega}_g - \boldsymbol{\nabla} \cdot \left(\phi \rho_g \boldsymbol{v}_g\right) \tag{5}$$

Although only a subset of the terms in the governing equations are verified in the current work, every term has been implemented in the code. Consequently, the material model and boundary conditions used in the code will be presented even though not all parts of the model are exercised in the problems presented here.

II.B. Physical Boundary Conditions

For most entry type ablation analyses, the incident heat flux is typically a combination of convective and radiative components. If a film coefficient boundary layer model is adopted where the species diffusion coefficients are assumed equal, and the mass and heat transfer film coefficients are assumed equal, then the energy balance for an infinitesimally thin control volume attached to the receding surface reduces to

$$\overbrace{\rho_e u_e C_H \left(h_r - h_w\right)}^{Convection} + \overbrace{\alpha \dot{q}_{rad}}^{Radiation} - \overbrace{\sigma \epsilon \left(T_w^4 - T_\infty^4\right)}^{Reradiation} - \overbrace{\dot{m}_w h_{o_w}}^{Wall\ Flux} - \overbrace{\dot{m}_c h_c}^{Char\ Flux} + \overbrace{\dot{m}_{g_s} h_{o_g}}^{Gas\ Flux} + \overbrace{\dot{q}_{cond_s}}^{Conduction} = 0 \tag{6}$$

and the surface mass balance is

$$\dot{m}_{g_s} - \dot{m}_w - \dot{m}_c = 0 \tag{7}$$

where

$$\text{Wall mass flux injected into fluid: } \dot{m}_w = \rho_w \boldsymbol{v}_w \cdot \hat{\boldsymbol{n}} \tag{8}$$

$$\text{Solid (char) flux to the surface from surface recession: } \dot{m}_c = \rho_c \dot{\boldsymbol{s}} \cdot \hat{\boldsymbol{n}} \tag{9}$$

$$\text{Pyrolysis gas flux to surface from in-depth convection and surface recession: } \dot{m}_{g_s} = \phi \rho_g \left(\boldsymbol{v}_g - \dot{\boldsymbol{s}}\right) \cdot \hat{\boldsymbol{n}} \tag{10}$$

$$\text{Conduction flux on solid side: } \dot{q}_{cond_s} = -k_s \boldsymbol{\nabla} T_s \cdot \hat{\boldsymbol{n}} \tag{11}$$

and $\hat{\boldsymbol{n}}$ is a unit normal vector pointing outward into the fluid. The boundary condition for the gas continuity equation is typically a specified pressure condition. Since there are no moving mesh ablation problems in the current study, several other simpler boundary conditions are verified. For the energy equation these include:

$$\text{Specified input heat flux: } \dot{q}_{spec} + \dot{q}_{cond_s} = 0 \tag{12}$$

$$\text{Perfect gas convection: } H\left(T_r - T_w\right) + \dot{q}_{cond_s} = 0 \tag{13}$$

and specified temperature. For the gas continuity equation, the boundary conditions verified in this study are specified pressure, specified density, and

$$\text{Specified inflow mass flux: } \dot{m}_{g_{spec}} + \dot{m}_{g_s} = 0 \tag{14}$$

Since there is no surface recession, the surface gas flux in Eq. (10) becomes

$$\dot{m}_{g_s} = \phi \rho_g \boldsymbol{v}_g \cdot \hat{\boldsymbol{n}} \tag{15}$$

The implementation of these boundary conditions in the finite element formulation will presented in a later section.

II.C. Material Model

In order to sufficiently explain the governing equations and boundary conditions, it is important to understand the material model used to characterize the state of the solid/gas mixture. It is assumed that all the pores are interconnected, and therefore the gas occupies all of the pore space and is free to flow through it. Consequently, the density of the solid/gas mixture is described by

$$\rho = \phi \rho_g + \rho_s \tag{16}$$

where the solid density is a bulk density, the gas density is a density with respect to the space the gas occupies (pore space), and the porosity is equal to the gas volume fraction. In terms of units, Eq. (16) can be expressed as

$$\overbrace{\frac{\text{[total mass]}}{\text{[total vol]}}}^{\rho} = \overbrace{\frac{\text{[pore vol]}}{\text{[total vol]}}}^{\phi} \overbrace{\frac{\text{[gas mass]}}{\text{[pore vol]}}}^{\rho_g} + \overbrace{\frac{\text{[solid mass]}}{\text{[total vol]}}}^{\rho_s} \tag{17}$$

It is assumed that the thermodynamic state of the pyrolysis gases can be described as a mixture of perfect gases, and that the solid and gas phases are in thermal equilibrium resulting in

$$T_g = T_s = T \tag{18}$$

$$P = f\left(\rho_g, T\right) \tag{19}$$

where the equation of state and all relationships describing the thermochemical state of the gas are dictated by an external equilibrium chemistry library such as Cantera[23] or CEA[24,25] to which the code is linked. During the code development effort, Cantera has been used to provide the thermochemical state of the gas. It should be noted that assuming that the gas is in thermochemical equilibrium may be a poor assumption for some problems. However the equilibrium assumption is being used for development purposes. Using a library such as Cantera does allow for easy extension to chemical non-equilibrium modeling.

The solid material model adopted in this study is similar to the model developed by Moyer and Rindal[1] but has been expanded to include an arbitrary number of components, nc. The solid bulk density is given by

$$\rho_s = \sum_{i=1}^{nc} \Gamma_i \rho_i \tag{20}$$

where Γ_i is the volume fraction of the i^{th} component in the virgin composite and is therefore constant. The units associated with the solid bulk density model in Eq. (20) are

$$\overbrace{\frac{\text{[solid mass]}}{\text{[total vol]}}}^{\rho_s} = \sum_{i=1}^{nc} \overbrace{\frac{\text{[initial vol of } i^{th} \text{ comp.]}}{\text{[total vol]}}}^{\Gamma_i} \overbrace{\frac{\text{[mass of } i^{th} \text{ comp.]}}{\text{[initial vol of } i^{th} \text{ comp.]}}}^{\rho_i} \tag{21}$$

It is assumed that the solid does not change volume due to thermal expansion, and therefore the total volume is constant. It is important to note that the solid description in Eq. (20) is only a modeling assumption, and the solid is not truly comprised of nc components, species, or distinguishable materials. This modeling assumption comes as a result of decomposition data obtained from thermogravimetric analyses (TGA).

It is assumed that all decomposed solid mass results in gas generation, and the general model of the decomposition process is described by the irreversible reaction

$$\text{virgin} \Longrightarrow \text{char} + \text{gas}$$

Taking the temporal derivative of Eq. (20) gives the solid decomposition rate in terms of component decomposition rates.

$$\frac{\partial \rho_s}{\partial t} = \sum_{i=1}^{nc} \Gamma_i \frac{\partial \rho_i}{\partial t} \tag{22}$$

It is assumed that the decomposition of each component can be described by an Arrhenius relationship of the form

$$\frac{\partial \rho_i}{\partial t} = -k_i \rho_{v_i} \left(\frac{\rho_i - \rho_{c_i}}{\rho_{v_i}} \right)^{m_i} e^{-E_i/RT} \tag{23}$$

which applies at a constant spatial location (as apposed to a given node which can move during the solution process).

Since most thermophysical properties of the solid are only known for the virgin and fully charred states, the intermediate solid is modeled as some interpolated state between virgin and char. This interpolated state is characterized by the extent of reaction (β), or degree of char, given by

$$\beta = \frac{\rho_v - \rho_s}{\rho_v - \rho_c} \tag{24}$$

where the virgin and char bulk densities are known constants. It is evident that as the solid decomposes from virgin to char, the extent of reaction ranges from 0 to 1. The definition in Eq. (24) can be rearranged to more clearly describe the interpolated state.

$$\rho_s = (1 - \beta)\, \rho_v + \beta \rho_c \tag{25}$$

Although the virgin and char materials are not distinguishable entities within the intermediate solid, Eq. (25) reveals that the degree of char represents an effective char volume fraction within the solid (not in the solid/gas mixture).

II.D. Property Models

II.D.1. Internal Energy and Enthalpy

The total internal energy and enthalpy of the system can be described by

$$\rho e_o = (1 - \beta)\, \rho_v e_v + \beta \rho_c e_c + \phi \rho_g e_{o_g} \tag{26}$$

and

$$\rho h_o = (1 - \beta)\, \rho_v h_v + \beta \rho_c h_c + \phi \rho_g h_{o_g} \tag{27}$$

where

$$e_{v/c} = h_{v/c} = h^o_{v/c} + \int_{T_o}^{T} C_{p_{v/c}} (T')\, dT' \tag{28}$$

and the total energy of the gas is

$$e_{o_g} = e_g + \frac{1}{2} (\boldsymbol{v}_g \cdot \boldsymbol{v}_g) \tag{29}$$

Similarly, the total enthalpy of the gas is

$$h_{o_g} = h_g + \frac{1}{2} (\boldsymbol{v}_g \cdot \boldsymbol{v}_g) \tag{30}$$

where the specific internal energy and specific enthalpy come from the chemistry library. Notice that the kinetic energy of the gas is not neglected in the current model in contrast to the CMA model. While it may be appropriate to neglect the kinetic energy for many ablation problems, it may not be appropriate for other porous flow problems that this code is intended to solve.

II.D.2. *Thermal Conductivity*

The thermal conductivity is in general an anisotropic function of temperature and pressure resulting in a dim-by-dim tensor where each component of the tensor can be an independent function of temperature. For materials with anisotropies not aligned with coordinate axes, an additional transformation matrix must be introduced to appropriately apply the model. The thermal conductivity of the mixture is assumed to be a mass weighted average of virgin char and gas.

$$\tilde{\boldsymbol{k}} = \frac{(1-\beta)\,\rho_v}{\rho}\tilde{\boldsymbol{k}}_v + \frac{\beta\rho_c}{\rho}\tilde{\boldsymbol{k}}_c + \frac{\phi\rho_g}{\rho}k_g\tilde{\boldsymbol{I}} \tag{31}$$

While more appropriate combination laws may exist to determine the mixture thermal conductivity, this simplified model has been chosen for development purposes.

II.D.3. *Permeability*

The permeability is in general an anisotropic function of the extent of reaction defined by Eq. (24) resulting in a dim-by-dim tensor where each component of the tensor can be an independent function of extent of reaction.

$$\tilde{\boldsymbol{\kappa}} = \tilde{\boldsymbol{\kappa}}\,(\beta) \tag{32}$$

Like the thermal conductivity, for materials with anisotropies not aligned with coordinate axes, an additional transformation matrix must be introduced to appropriately apply the model.

II.D.4. *Porosity*

The porosity is in general a function of extent of reaction, and represents the gas mass fraction since all of the pore space is assumed to be interconnected.

$$\phi = \phi\,(\beta) \tag{33}$$

II.D.5. *Emissivity and Absorptivity*

The emissivity and absorptivity of the mixture are assumed to be equal to the emissivity and absorptivity of the solid. They are modeled similarly to the thermal conductivity while only taking the solid portion into account

$$\epsilon = \frac{(1-\beta)\,\rho_v}{\rho_s}\epsilon_v + \frac{\beta\rho_c}{\rho_s}\epsilon_c \tag{34}$$

and

$$\alpha = \frac{(1-\beta)\,\rho_v}{\rho_s}\alpha_v + \frac{\beta\rho_c}{\rho_s}\alpha_c \tag{35}$$

II.E. Porous Flow Law

Applying the Navier-Stokes momentum equations to flow through the char layer would require detailed knowledge of the pore structure, and that information is typically not known. Consequently, a porous flow law can be used as simplified momentum equations. Porous flow laws typically require extra knowledge about the material beyond thermophysical properties. These properties can include the porosity and permeability of the solid, as well as the viscosity of the gas flowing through the porous medium. The porosity and permeability can be determined through material testing and are provided to the code as a function of extent of reaction. For development purposes, Darcy's law[26] is being used to govern the porous flow. Through experiments, Darcy determined how the volumetric flow rate of a laminar flowing fluid is related to the pressure gradient within the fully saturated porous medium.

$$\boldsymbol{Q} = -A\frac{\widetilde{\boldsymbol{\kappa}}}{\mu}\boldsymbol{\nabla}P \tag{36}$$

The superficial or filtration velocity is the volumetric flow rate averaged over the cross-sectional area of the medium and is given by

$$\boldsymbol{v}'_g = \frac{\boldsymbol{Q}}{A} = -\frac{\widetilde{\boldsymbol{\kappa}}}{\mu}\boldsymbol{\nabla}P \tag{37}$$

The average or seepage velocity of the fluid is the volumetric flow rate averaged over the cross-sectional area through which the fluid can flow (porous area) and is given by

$$\boldsymbol{v}_g = \frac{\boldsymbol{Q}}{\phi A} = -\frac{\widetilde{\boldsymbol{\kappa}}}{\phi \mu}\boldsymbol{\nabla} P \tag{38}$$

which assumes that the surface porosity is equal to the volumetric porosity. Darcy's law is valid for steady laminar flows with "sufficiently" low Reynolds numbers. While it is conceded that the pyrolysis gas flow within the porous ablator may not be adequately described by Darcy's law for many situations, it is being used for code development purposes due to its simplicity. The addition of more appropriate physical models can certainly be accomplished once sufficient data is available to support such a change and the necessary material properties are known.

The seepage velocity can be used to determine the gas mass flux (mass flow rate per unit area) at any point within the medium according to

$$\dot{m}_g = (\phi \rho_g)\, \boldsymbol{v}_g = -\,(\phi \rho_g)\, \frac{\widetilde{\boldsymbol{\kappa}}}{\phi \mu}\boldsymbol{\nabla} P = -\rho_g \frac{\widetilde{\boldsymbol{\kappa}}}{\mu}\boldsymbol{\nabla} P = \rho_g \boldsymbol{v}_g' \tag{39}$$

Each individual velocity component can be calculated according to

$$(u_g)_k = \frac{-1}{\phi \mu}\widetilde{\boldsymbol{\kappa}}_k \cdot \boldsymbol{\nabla} P \tag{40}$$

where k denotes the component and $\widetilde{\boldsymbol{\kappa}}_k$ denotes the k^{th} row vector of the permeability tensor. Substituting Darcy's law into Eq. (3) gives

$$\frac{\partial(\phi \rho_g)}{\partial t} = \dot{\omega}_g + \boldsymbol{\nabla} \cdot \left(\rho_g \frac{\widetilde{\boldsymbol{\kappa}}}{\mu}\boldsymbol{\nabla} P\right) + \boldsymbol{\nabla} \cdot (\phi \rho_g \boldsymbol{v}_m) \tag{41}$$

Notice that Darcy's law models the convective term as a diffusive term analogous to Fourier's law for heat conduction and Fick's law for diffusion. For stationary mesh problems, this results in a character change from hyperbolic to parabolic which aids in the stability of the solution procedure.

II.F. Galerkin Weak Statement

The governing equations can be restated as

$$\text{Energy: } \frac{\partial(\rho e_o)}{\partial t} - \boldsymbol{\nabla} \cdot \left(\widetilde{\boldsymbol{k}}\boldsymbol{\nabla} T\right) + \boldsymbol{\nabla} \cdot \left(\phi \rho_g h_{o_g} \boldsymbol{v}_g\right) - \boldsymbol{\nabla} \cdot (\rho h_o \boldsymbol{v}_m) - \dot{Q} = 0 \tag{42}$$

$$\text{Solid Mass: } \frac{\partial \rho_s}{\partial t} - \dot{\omega}_s - \boldsymbol{\nabla}\cdot (\rho_s \boldsymbol{v}_m) = 0 \tag{43}$$

and

$$\text{Gas Mass: } \frac{\partial(\phi \rho_g)}{\partial t} - \dot{\omega}_g + \boldsymbol{\nabla} \cdot (\phi \rho_g \boldsymbol{v}_g) - \boldsymbol{\nabla} \cdot (\phi \rho_g \boldsymbol{v}_m) = 0 \tag{44}$$

where

$$\text{Darcy: } \boldsymbol{v}_g = -\frac{\widetilde{\boldsymbol{\kappa}}}{\phi \mu}\boldsymbol{\nabla} P \tag{45}$$

Since the solid mass equation on a stationary grid only depends on the temperature and density history of a given point, it is acceptable to solve the solid mass equation on a stationary mesh while solving the other equations on a moving mesh. Consequently, the solution procedure will be to solve for the temperature and gas density fields through integration of the partial differential equations (PDEs) given in Eqs. (42) and (44) while the solid density will be treated as a nonlinearity, and the resulting ordinary differential equation (ODE) governing the solid density evolution will be numerically integrated.

II.F.1. Energy Equation

A Galerkin weak statement can be developed for the energy equation (Eq. (42)) by first multiplying it by a suitable test function, v, and integrating over the domain, Ω with boundary, Γ, while integrating the second, third, and fourth terms by parts to give the natural boundary condition terms. The weak statement is then: Find $T \in H^1$ such that

$$\int_\Omega \left[v\frac{\partial(\rho e_o)}{\partial t} + \boldsymbol{\nabla} v \cdot \left(\widetilde{\boldsymbol{k}}\boldsymbol{\nabla} T\right) - \boldsymbol{\nabla} v \cdot \left(\phi\rho_g h_{o_g} \boldsymbol{v}_g\right) + \boldsymbol{\nabla} v \cdot \left(\rho h_o \boldsymbol{v}_m\right) - v\dot{Q} \right] d\Omega \\ + \oint_\Gamma \left(v h_{o_g} \dot{m}_g - v h_o \dot{m}_m + v \dot{q}_{cond_s}\right) d\Gamma = 0 \qquad \forall v \in H_0^1 \tag{46}$$

where the boundary mass flux due to gas convection is

$$\dot{m}_g = (\phi\rho_g)\,\boldsymbol{v}_g \cdot \hat{\boldsymbol{n}} \tag{47}$$

the boundary mass flux due to mesh motion is

$$\dot{m}_m = \rho \boldsymbol{v}_m \cdot \hat{\boldsymbol{n}} \tag{48}$$

II.F.2. Gas Mass Conservation Equation

Likewise, a Galerkin weak statement can be developed for the gas mass conservation equation,Eq. (44). Again, the equation will be multiplied by a suitable test function and integrated over the domain while integrating the third and fourth terms by parts to give the natural boundary condition terms. The weak statement is then: Find $\phi\rho_g \in H^1$ such that

$$\int_\Omega \left(\frac{\partial(\phi\rho_g)}{\partial t} v - \boldsymbol{\nabla} v \cdot (\phi\rho_g \boldsymbol{v}_g) + \boldsymbol{\nabla} v \cdot (\phi\rho_g \boldsymbol{v}_m) + \dot{m}_s v \right) d\Omega + \oint_\Gamma \left[v\left(\dot{m}_g - \dot{m}_{g_m}\right)\right] d\Gamma = 0 \qquad \forall v \in H_0^1 \tag{49}$$

where the pyrolysis gas mass flux at the boundary due to mesh motion is

$$\dot{m}_{g_m} = (\phi\rho_g)\,\boldsymbol{v}_m \cdot \hat{\boldsymbol{n}} \tag{50}$$

II.G. Finite Element Formulation

Eqs. (46) and (49) can be discretized by expanding the independent variables and test functions in terms of a finite dimensional basis

$$\left.\begin{aligned} (\rho e_o)_h(\boldsymbol{x}) &= \sum_{j=1}^{N} (\rho e_o)_j \,\psi_j(\boldsymbol{x}) & (\rho h_o)_h(\boldsymbol{x}) &= \sum_{j=1}^{N} (\rho h_o)_j \,\psi_j(\boldsymbol{x}) \\ (\phi\rho_g)_h(\boldsymbol{x}) &= \sum_{j=1}^{N} (\phi\rho_g)_j \,\psi_j(\boldsymbol{x}) & T_h(\boldsymbol{x}) &= \sum_{j=1}^{N} T_j \psi_j(\boldsymbol{x}) \\ \boldsymbol{\nabla} T_h(\boldsymbol{x}) &= \sum_{j=1}^{N} T_j \boldsymbol{\nabla}\psi_j(\boldsymbol{x}) & \boldsymbol{\nabla} P_h(\boldsymbol{x}) &= \sum_{j=1}^{N} P_j \boldsymbol{\nabla}\psi_j(\boldsymbol{x}) \\ v_h(\boldsymbol{x}) &= \sum_{i=1}^{N} v_i \psi_i(\boldsymbol{x}) & \boldsymbol{\nabla} v_h(\boldsymbol{x}) &= \sum_{i=1}^{N} v_i \boldsymbol{\nabla}\psi_i(\boldsymbol{x}) \end{aligned}\right\} \tag{51}$$

where the subscript h is introduced to denote a finite dimensional approximation. In this study, first order Lagrangian basis functions are used to give a second order accurate spatial discretization, and the verification of this order of accuracy will be demonstrated in Section III. Notice that ρe_o, ρh_o, and T behave according to the same finite dimensional basis. While this may introduce physical inconsistencies within and element, the errors vanish as the mesh is refined. This was chosen to simplify implementation and to improve computational efficiency. This assumption will be shown to not affect the spatial order of accuracy of the scheme. During the solution process, ρe_o and ρh_o will be evaluated at the nodes with the known nodal temperatures and densities, and all four quantities will be interpolated to the element interior using the basis functions. An analogous situation exists for the pressure gradient with respect to temperature and density.

Since the unknowns are no longer functions of $\boldsymbol{x}$, the PDE system reduces to an ODE system in which the temporal derivatives can be defined as

$$\frac{\partial(\rho e_o)_h}{\partial t} = \sum_{j=1}^{N} \frac{d\left(\rho e_o\right)_j}{dt} \psi_j\left(\boldsymbol{x}\right) \tag{52}$$

$$\frac{\partial(\phi \rho_g)_h}{\partial t} = \sum_{j=1}^{N} \frac{d\left(\phi \rho_g\right)_j}{dt} \psi_j\left(\boldsymbol{x}\right) \tag{53}$$

Since the equation system should be satisfied for all combinations of nodal shape function coefficients, v_i, their choice is arbitrary as long as a unique combination is chosen for each node.[27] In typical finite element fashion, the coefficients are chosen to be $v_i = \delta_{il}$ for the l^{th} nodal equation. Consequently, the nodal residual equations resulting from Eqs. (46) and (49) can now be written

$$\int_\Omega \psi_i \frac{\partial(\rho e_o)_h}{\partial t} d\Omega + \int_\Omega \boldsymbol{\nabla}\psi_i \cdot \widetilde{\boldsymbol{k}} \boldsymbol{\nabla} T_h d\Omega - \int_\Omega \left(\phi\rho_g\right)_h h_{o_g} \boldsymbol{\nabla}\psi_i \cdot \boldsymbol{v}_g d\Omega$$
$$+ \int_\Omega \left(\rho h_o\right)_h \boldsymbol{\nabla}\psi_i \cdot \boldsymbol{v}_m d\Omega + \int_\Gamma \psi_i \dot{q}_{cond_s} d\Gamma + \int_\Gamma \psi_i h_{o_g} \dot{m}_g d\Gamma - \int_\Gamma \psi_i h_o \dot{m}_m d\Gamma - \int_\Omega \psi_i \dot{Q} d\Omega = 0 \tag{54}$$

and

$$\int_\Omega \psi_i \frac{\partial(\phi\rho_g)_h}{\partial t} d\Omega - \int_\Omega \left(\phi\rho_g\right)_h \boldsymbol{\nabla}\psi_i \cdot \boldsymbol{v}_g d\Omega + \int_\Omega \left(\phi\rho_g\right)_h \boldsymbol{\nabla}\psi_i \cdot \boldsymbol{v}_m d\Omega$$
$$+ \int_\Gamma \psi_i \dot{m}_g d\Gamma - \int_\Gamma \psi_i \dot{m}_{g_m} d\Gamma + \int_\Omega \psi_i \dot{m}_s d\Omega = 0 \tag{55}$$

for i = 1,2,...,N.

II.H. Time Discretization

The semidiscrete weak form of the system given by Eqs. (54) and (55) is discretized in time using backwards finite difference schemes. Both first and second-order accurate schemes may be derived from Taylor series expansions in time about $\boldsymbol{U}_h\left(t_{n+1}\right) = \boldsymbol{U}_{n+1}$:

$$\boldsymbol{U}_n = \boldsymbol{U}_{n+1} + \frac{\partial \boldsymbol{U}_{n+1}}{\partial t}\left(t_n - t_{n+1}\right) + \frac{\partial^2 \boldsymbol{U}_{n+1}}{\partial t^2}\frac{\left(t_n - t_{n+1}\right)^2}{2} + \mathcal{O}\left(\left(t_n - t_{n+1}\right)^3\right)$$
$$\boldsymbol{U}_{n-1} = \boldsymbol{U}_{n+1} + \frac{\partial \boldsymbol{U}_{n+1}}{\partial t}\left(t_{n-1} - t_{n+1}\right) + \frac{\partial^2 \boldsymbol{U}_{n+1}}{\partial t^2}\frac{\left(t_{n-1} - t_{n+1}\right)^2}{2} + \mathcal{O}\left(\left(t_{n-1} - t_{n+1}\right)^3\right)$$

These expressions can be manipulated to create difference formulas of the form[28,29]

$$\frac{\partial \boldsymbol{U}_{n+1}}{\partial t} = \alpha_t \boldsymbol{U}_{n+1} + \beta_t \boldsymbol{U}_n + \gamma_t \boldsymbol{U}_{n-1} + \mathcal{O}\left(\Delta t_{n+1}^p\right) \tag{56}$$

to yield either a first or second-order accurate scheme. The weights α_t, β_t, and γ_t are given for $p = 1$ and $p = 2$ in Table 1.

Table 1. First and second-order accurate time discretization coefficients.

$\boldsymbol{p}$	$\boldsymbol{\alpha_t}$	$\boldsymbol{\beta_t}$	$\boldsymbol{\gamma_t}$
1	$\frac{1}{\Delta t_{n+1}}$	$\frac{-1}{\Delta t_{n+1}}$	0
2	$-\beta_t - \gamma_t$	$-\left[\frac{1}{\Delta t_{n+1}} + \frac{1}{\Delta t_n}\right]$	$\frac{\Delta t_{n+1}}{\Delta t_n\left(\Delta t_{n+1} + \Delta t_n\right)}$

The resulting temporal derivatives are

$$\frac{d\left(\rho e_o\right)_j}{dt} = \alpha_t \left(\rho e_o\right)_j^{n+1} + \beta_t \left(\rho e_o\right)_j^{n} + \gamma_t \left(\rho e_o\right)_j^{n-1} \tag{57}$$

and

$$\frac{d\left(\phi\rho_g\right)_j}{dt} = \alpha_t \left(\phi\rho_g\right)_j^{n+1} + \beta_t \left(\phi\rho_g\right)_j^{n} + \gamma_t \left(\phi\rho_g\right)_j^{n-1} \tag{58}$$

II.I. Linearization with the Complex-Step Method

Eqs. (54) and (55) form a discrete nonlinear system of equations with unknown nodal values T^{n+1} and $(\phi\rho_g)^{n+1}$. This nonlinear system will be solved with a typical Newton scheme. Since the intent is to reach a fully implicit formulation, it is necessary to linearize the residual equations in iteration space. This will be done according to the familiar Taylor-series expansion

$$\mathcal{R}_i^{\nu+1} = \mathcal{R}_i^{\nu} + \sum_{j=1}^{N} \left\{ \frac{\partial \mathcal{R}_i}{\partial T_j}\bigg|^{\nu} \Delta T_j + \frac{\partial \mathcal{R}_i}{\partial (\phi\rho_g)_j}\bigg|^{\nu} \Delta\left(\phi\rho_g\right)_j \right\} + \text{higher order terms} \tag{59}$$

where the superscript ν has been introduced to denote iteration level and

$$\Delta T_j = T_j^{\nu+1} - T_j^{\nu} \tag{60}$$

and

$$\Delta\left(\phi\rho_g\right)_j = \left(\phi\rho_g\right)_j^{\nu+1} - \left(\phi\rho_g\right)_j^{\nu} \tag{61}$$

Keep in mind that the Jacobian terms, $\frac{\partial \mathcal{R}_i}{\partial ()_j}$, are only nonzero if the support of the basis functions ψ_i and ψ_j overlap.

Since the intent of the code is to allow for easy addition of material models and porous flow laws, the Jacobians will not be derived analytically, rather they will be calculated exactly (to machine precision) using the complex-step method. The implementation of the complex-step method does not disallow the use of analytically derived Jacobian terms (or some mix of the two), so these terms can be replaced with analytical expressions at a later date if desired.

The residuals can be expanded according to a Taylor-series expansion in independent variable space about a point $\left[T_j, \left(\phi\rho_g\right)_j\right]$. For example, an expansion with complex step in temperature, $i\Delta T$, is

$$\mathcal{R}\left[T_j + i\Delta T_j, \left(\phi\rho_g\right)_j\right] = \mathcal{R}\left[T_j, \left(\phi\rho_g\right)_j\right] + i\frac{\partial \mathcal{R}}{\partial T_j}\left(\Delta T_j\right) - \frac{1}{2}\frac{\partial^2 \mathcal{R}}{\partial T_j^2}\left(\Delta T_j\right)^2 - \frac{i}{6}\frac{\partial^3 \mathcal{R}}{\partial T_j^3}\left(\Delta T_j\right)^3 + \text{higher order terms} \tag{62}$$

Now looking at just the imaginary part and ignoring the higher order terms gives

$$\Im\left\{\mathcal{R}\left[T_j + i\Delta T_j, \left(\phi\rho_g\right)_j\right]\right\} = \frac{\partial \mathcal{R}}{\partial T_j}\Delta T_j - \frac{1}{6}\frac{\partial^3 \mathcal{R}}{\partial T_j^3}\left(\Delta T_j\right)^3 \tag{63}$$

Solving for the first derivative gives the Jacobian terms

$$\frac{\partial \mathcal{R}}{\partial T_j} = \frac{\Im\left\{\mathcal{R}\left[T_j + i\Delta T_j, \left(\phi\rho_g\right)_j\right]\right\}}{\Delta T_j} + \mathcal{O}\left[\left(\Delta T_j\right)^2\right] \tag{64}$$

This process can be repeated for each independent variable. The proper choice for the perturbation steps is not immediately obvious from looking at the equations. In general, the perturbation step should have some relationship to the order of magnitude of the independent variable (in this case T_j).

$$\Delta T_j = rT_j \tag{65}$$

Taking advantage of the fact that the problem will be solved on a finite precision machine, r can be chosen so that the second order error term in the Jacobian expression in Eq. (64) is smaller than the smallest order represented in the first term. Consequently, by taking advantage of truncation error, the derivatives will be accurate to machine precision. Having such a small step does not affect the division operation in the first term of Eq. (64) because multiplication/division operations simply result in an exponent shift. $r = 10^{-20}$ is used for the calculations in this study, and an investigation to the sensitivity of this parameter will be presented in Section IV.

A complex residual calculation is required for each Jacobian term. The scope of the residual recalculations can be reduced by only recalculating those terms that depend on the degree of freedom with the current complex perturbation. One fortuitous aspect of this method is that there is no separate residual calculation required for the unperturbed state (Right Hand Side of linear system). If only one independent variables is perturbed (use T for example), then the real part of Eq. (62) can be solved for the residual

$$\mathcal{R}\left[T_j, \left(\phi\rho_g\right)_j\right] = \Re\left\{\mathcal{R}\left[T_j + i\Delta T_j, \left(\phi\rho_g\right)_j\right]\right\} + \mathcal{O}\left[\left(\Delta T_j\right)^2\right] \tag{66}$$

Again since the independent variable perturbation has been chosen to be sufficiently small, the order of magnitude of the error terms is smaller than the smallest order represented in the complex residual calculation. This results in a residual calculation exact to machine precision. Through calculation of the Jacobian terms with the complex step method, the calculation of the real residual comes free.

The advantage of the complex-step method is in the development cost. The derivation, implementation, and debugging of analytical Jacobian terms can be time consuming. Plus, the addition and/or modification of models with the complex-step method can be accomplished with relative ease. Additionally, most codes with complicated governing equations employ approximate Jacobians which will result in less the optimal convergence of the nonlinear solution routine. The drawback of the complex-step method is the extra time spent doing complex arithmetic. While more memory is required to store complex variables, the necessity for complex variables only occurs at the element assembly level. Therefore the memory requirements for the complex-step method are insignificant relative to the memory requirements for storing the mesh, variables at several time levels, and the linear system.

II.J. Additional Numerical Issues

The integrals in Eqs. (54) and (55) are evaluated numerically with using Gaussian quadrature, and the resulting linear system is solved using the Generalized Minimal Residual method with preconditioning.[21] The code was developed using the libMesh finite element library.[30] Essentially, the library handles all of the physics independent aspects of the code including I/O, mesh connectivity and element data, domain decomposition, parallel communication, adaptive mesh refinement, and linear system solution. libMesh provides a seamless interface to several other libraries including METIS[31] for domain decomposition and PETSc[32] for solving the linear system.

II.K. Boundary Conditions

The code can handle a rich suite of boundary conditions, including multiple flux type boundary conditions that can be summed over a given boundary. Only those that are used in this study will be described here.

II.K.1. Specified Temperature

Dirichlet conditions and other conditions that require the substitution of a nodal residual equation with a boundary condition specific equation will be imposed via the penalty boundary method (PBM).[33] With this method, the L^2 projection of the residuals are added to the matrix. The projection is multiplied by some large factor so that, in floating point arithmetic, the existing (smaller) entries in the matrix and right-hand-side are effectively ignored. The boundary integral in the weak form becomes

$$\frac{1}{\epsilon}\int_{\Gamma}\psi_i \mathcal{R} d\Gamma = 0 \tag{67}$$

where $\epsilon \ll 1$. For the specified temperature condition, the residual equation is given by

$$\mathcal{R} = T_{spec} - T_h\left(\boldsymbol{x}\right) = 0 \tag{68}$$

where T_{spec} is the known specified temperature. Substituting in the definition of $T_h\left(\boldsymbol{x}\right)$, Eq. (51), gives

$$\mathcal{R} = T_{spec} - \sum_{j=1}^{N} T_j \psi_j\left(\boldsymbol{x}\right) = 0 \tag{69}$$

Substituting the residual definition into Eq. (67) gives the weak form of the discrete residual equations

$$\mathcal{R}_i = \frac{1}{\epsilon}\int_{\Gamma}\psi_i\left(T_{spec} - \sum_{j=1}^{N} T_j \psi_j\left(\boldsymbol{x}\right)\right) d\Gamma = 0 \tag{70}$$

Employing a numerical integration technique, such as Gaussian quadrature, to evaluate the surface integral gives

$$\mathcal{R}_i = \frac{1}{\epsilon}\sum_{QP=1}^{\#\,\text{QPs}}\left[J \times W\psi_i\left(T_{spec} - \sum_{j=1}^{N} T_j \psi_j\right)\right]\Bigg|_{QP} \tag{71}$$

where QP denotes the quadrature point and $J \times W$ is the Jacobian of the transformation matrix times the integration weight. Linearizing in iteration space according to a Taylor-series expansion while ignoring higher order terms gives

$$\mathcal{R}_i^{\nu+1} = \mathcal{R}_i^{\nu} + \sum_{j=1}^{N} \left.\frac{\partial \mathcal{R}_i}{\partial T_j}\right|^{\nu} \Delta T_j \tag{72}$$

Differentiating Eq. (71) gives the Jacobian terms for the linear system

$$\frac{\partial \mathcal{R}_i}{\partial T_j} = -\frac{J \times W}{\epsilon} \psi_i \psi_j \tag{73}$$

Given the ease of the Jacobian derivation, this boundary condition has been implemented using analytical Jacobians as opposed to using the complex perturbation approach.

II.K.2. Specified Heat Flux

The specified flux boundary condition is linear and requires no Jacobian contributions. The boundary heat flux term in Eq. (54) becomes

$$\int_{\Gamma} \psi_i \dot{q}_{cond_s} d\Gamma = -\int_{\Gamma} \psi_i \dot{q}_{spec} d\Gamma \tag{74}$$

where q_{spec} is the known specified flux. Note that the sign change is intended to simplify usability. In the original PDE derivation, fluxes into the body were considered a negative quantity. So the sign change allows the user to input a positive heat flux when the intent is to put heat into the body. Introducing Gaussian quadrature for numerical integration gives

$$-\int_{\Gamma} \psi_i \dot{q}_{spec} d\Gamma = \sum_{QP=1}^{\#\text{ QPs}} -J \times W \dot{q}_{spec} \psi_i \tag{75}$$

II.K.3. Convection

The convective heat flux is given by

$$\dot{q}_{conv} = H\left(T_r - T_w\right) \tag{76}$$

Substituting into the boundary heat flux term in Eq. (54) gives

$$\int_{\Gamma} \psi_i \dot{q}_{cond_s} d\Gamma = -\int_{\Gamma} \psi_i H\left(T_r - T_w\right) d\Gamma \tag{77}$$

Introducing Gaussian quadrature for numerical integration and the finite dimensional basis representation of temperature gives

$$-\int_{\Gamma} \psi_i H\left(T_r - T_w\right) d\Gamma = \sum_{QP=1}^{\#\text{ QPs}} \left.\left[-J \times W \psi_i H \left(T_r - \sum_{j=1}^{N} T_j \psi_j\left(\boldsymbol{x}\right)\right)\right]\right|_{QP} \tag{78}$$

The Jacobian contributions for the convective boundary condition are calculated using the complex perturbation method.

II.K.4. Specified Gas Density

The specified gas density condition is a Dirichlet condition on the gas continuity equation. In real world problems, the surface density is not likely known. Therefore it is expected that this condition will be infrequently used. However, it is very useful for debugging purposes and its implementation is verified in this study. The specified gas density condition is imposed via PBM, and the residual equation is given by

$$\mathcal{R} = \left(\phi\rho_g\right)_{spec} - \left(\phi\rho_g\right)_h\left(\boldsymbol{x}\right) = 0 \tag{79}$$

where $\left(\phi\rho_g\right)_{spec}$ is the known specified density. Substituting in the definition of $\left(\phi\rho_g\right)_h\left(\boldsymbol{x}\right)$, Eq. (51), gives

$$\mathcal{R} = \left(\phi\rho_g\right)_{spec} - \sum_{j=1}^{N} \left(\phi\rho_g\right)_j \psi_j\left(\boldsymbol{x}\right) = 0 \tag{80}$$

Substituting the residual definition into Eq. (67) gives the weak form of the discrete residual equations

$$\mathcal{R}_i = \frac{1}{\epsilon}\int_\Gamma \psi_i \left((\phi\rho_g)_{spec} - \sum_{j=1}^{N} (\phi\rho_g)_j \, \psi_j(\boldsymbol{x}) \right) d\Gamma = 0 \tag{81}$$

Employing a numerical integration technique, such as Gaussian quadrature, to evaluate the surface integral gives

$$\mathcal{R}_i = \frac{1}{\epsilon}\sum_{QP=1}^{\#\text{ QPs}} \left[J \times W\psi_i \left((\phi\rho_g)_{spec} - \sum_{j=1}^{N} (\phi\rho_g)_j \, \psi_j \right) \right] \Bigg|_{QP} \tag{82}$$

and differentiating Eq. (82) gives the Jacobian terms for the linear system

$$\frac{\partial \mathcal{R}_i}{\partial (\phi\rho_g)_j} = -\frac{J \times W}{\epsilon}\psi_i\psi_j \tag{83}$$

Given the ease of the Jacobian derivation, this boundary condition has been implemented using analytical Jacobians as opposed to using the complex perturbation approach. Although the gas density degree of freedom is specified with this condition, the energy equation still has a pyrolysis gas mass flux term that must be evaluated along with its Jacobians.

II.K.5. Specified Mass Flux

The specified mass flux condition effects terms in both the energy and gas continuity equations.

$$\text{Energy: } \int_\Gamma \psi_i h_{o_g} \dot{m}_w d\Gamma = -\int_\Gamma \psi_i h_{o_g} \dot{m}_{spec} d\Gamma \tag{84}$$

and

$$\text{Gas Mass: } \int_\Gamma \psi_i \dot{m}_w d\Gamma = -\int_\Gamma \psi_i \dot{m}_{spec} d\Gamma \tag{85}$$

Like the specified heat flux condition, the negative sign is introduced so that the user inputs a positive mass flux to denote flow into the body. This condition is linear for the gas continuity equation, however it is nonlinear for the energy equation since the gas total enthalpy depends on both density and temperature. Consequently, Jacobian terms are required, and these again are calculated through the complex-perturbation method.

II.K.6. Specified Pressure

Since the independent variables in the linear system are temperature and gas density, the specified pressure condition is nonlinear and must be applied through PBM to replace the appropriate gas density residual equations in the linear system. The residual equation for the specified pressure condition is

$$\mathcal{R} = P_{spec} - P(\boldsymbol{x}) = 0 \tag{86}$$

where P_{spec} is the known specified pressure. Substituting the residual into Eq. (67) gives the weak form of the discrete residual equations

$$\mathcal{R}_i = \frac{1}{\epsilon}\int_\Gamma \psi_i (P_{spec} - P) \, d\Gamma = 0 \tag{87}$$

Again using Gaussian quadrature to evaluate the surface integral gives

$$\mathcal{R}_i = \frac{1}{\epsilon}\sum_{QP=1}^{\#\text{ QPs}} \left[J \times W\psi_i (P_{spec} - P) \right]\big|_{QP} \tag{88}$$

The Jacobian entries are calculated using the complex-perturbation method. The energy equation still has a pyrolysis gas mass flux term that must be evaluated along with its Jacobians.

III. Verification

Verification exercises are typically done by generating solutions on successively refined space or time meshes for problems with known exact solutions. Examining the solution error as the mesh is refined shows the order of accuracy of the model as implemented. Typically, the spatial and temporal orders of accuracy are independently verified. If the discretization error is of the form

$$DE = a\Delta t^{p} + b\Delta h^{q} + \text{(higher order terms)} \tag{89}$$

Then the temporal order of accuracy can be determined by making $b\Delta h^{q}$ negligible through the use of a very fine mesh and solving a problem with successively refined time steps. The logarithm of the discretization error as a function of the logarithm of the time step should have a slope of p

$$\frac{\partial[\log{(DE)}]}{\partial[\log{(\Delta t)}]} = p$$

with intercept $\log{(a)}$. Similarly for the spatial order of accuracy, $a\Delta t^{p}$ can be made negligibly small through the use of an extremely small time step. The logarithm of the discretization error as a function of the logarithm of the mesh spacing should have a slope of q

$$\frac{\partial[\log{(DE)}]}{\partial[\log{(\Delta h)}]} = q$$

with intercept $\log{(b)}$. This method can be time consuming considering that the problems either need to be solved with an extremely fine mesh or time step.

In the current study, an alternative approach is considered in which the spatial and temporal orders of accuracy are simultaneously verified.[15] Factoring out a $b\Delta h^{q}$ from the discretization error gives

$$DE = \left(1 + \frac{a\Delta t^{p_{th}}}{b\Delta h^{q_{th}}}\right) b\Delta h^{q_{th}} \tag{90}$$

where the subscript th is introduced to denote the theoretical orders of accuracy. If it is assumed that the theoretical orders of accuracy p_{th} and q_{th} are known, which they usually are, then simultaneous refinement of the spatial and temporal meshes while holding $\frac{a\Delta t^{p_{th}}}{b\Delta h^{q_{th}}}$ constant can be performed. If the observed slope of the log-log relationship between the discretization error and mesh spacing, q_{obs}, is the same as the expected theoretical slope, q_{th},

$$\frac{\partial[\log{(DE)}]}{\partial[\log{(\Delta h)}]} = q_{th} = q_{obs} \tag{91}$$

with intercept

$$\log\left[b\left(1 + \frac{a\Delta t^{p_{th}}}{b\Delta h^{q_{th}}}\right)\right]$$

then both the temporal and spatial orders of accuracy are simultaneously confirmed as long as $\frac{a\Delta t^{p}}{b\Delta h^{q}}$ is on the order of or greater than 1. This is because one would like the intercept to be sensitive to this quantity so that a non-co-linear result would suggest that either p_{obs} or q_{obs} is not what is expected from theory. The drawback of this methodology is that if the expected result is not produced, then the developer cannot be certain whether the error is in the temporal or spatial discretization. However, if the implementation of the temporal discretization has been previously verified, then the error would have to lie in the spatial discretization. This method offers a significant time savings over the traditional independent refinement approach since it is not necessary to solve the problem with extremely small time steps or fine meshes.

When comparing a numerical solution to an exact solution there are several different acceptable forms of the discretization error metric. These could include the value of independent variables at a specific location and/or time, or the metric could be global comparison of the solutions. For this study, the discrete normalized root-mean squared (RMS) error of a nodal quantity of interest across the domain is chosen as the error metric.

$$DE_{RMS} = \frac{1}{Q_o}\sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(Q_{exact} - Q_{sim}\right)^2} \tag{92}$$

where N denotes the number of nodes, $exact$ denotes the exact solution, sim denotes the simulated solution, Q denotes the quantity of interest (usually the independent variable(s)), and Q_o represents the normalization value (usually the initial value). When evaluating the error as a function of mesh refinement, the grid metric has been chosen to be

$$\text{Number of Elements in One Dimension: } \sqrt[dim]{N_e}$$

where dim is the number of spatial dimensions in the problem and N_e is the total number of elements in the domain. The number of elements in one dimension is inversely proportional to the characteristic mesh length scale, Δh. Since for a given problem, the characteristic domain length, L, is constant, the characteristic mesh length scale can be defined as

$$\Delta h = \frac{L}{\sqrt[dim]{N_e}} \tag{93}$$

Substituting this relationship into Eq. (90) gives

$$DE_{RMS} = \left(1 + \frac{a\Delta t^{p_{th}}}{b\Delta h^{q_{th}}}\right) bL^{q_{th}} \left(\sqrt[dim]{N_e}\right)^{-q_{th}} \tag{94}$$

This leads to the verification criteria

$$\frac{\partial[\log\left(DE_{RMS}\right)]}{\partial\left[\log\left(\sqrt[dim]{N_e}\right)\right]} = -q_{th} = -q_{obs} \tag{95}$$

This means that in order to simultaneously confirm the spatial and temporal orders of accuracy, a slope of $-q^{th}$ in $\log\left(DE_{RMS}\right)$-$\log\left(\sqrt[dim]{N_e}\right)$ space must be observed when solving the problems on successively refined space and time meshes while holding $\frac{a\Delta t^{p_{th}}}{b\Delta h^{q_{th}}}$ constant.

In this study, second order spatial accuracy ($q = 2$) and second order temporal accuracy ($p = 2$) will be verified through the simultaneous space/time refinement approach outlined above. Additionally, it is the intent to verify both the two and three dimensional capabilities for all allowable element types. The element types included in this study are: triangles and quadrilaterals in two dimensions and hexahedrons, tetrahedrons, and prisms in three dimensions. While verification studies have been performed with pyramids, none of the problems included in this study were solved with pyramid elements.

In addition to verifying the temporal and spatial orders of accuracy, it is necessary to check the rate of convergence for the Newton scheme. Theoretically, a correct implementation of Newton's method should exhibit second-order convergence. If the solution converges at some other rate, it is likely that a mistake was made in the derivation and/or implementation of the method, likely in the Jacobian terms. The theoretical relationship between error, E, at successive iterations is given by

$$E^{\nu+1} = a\left(E^{\nu}\right)^2 \tag{96}$$

In this study, the relative L^2 of the independent variable correction vector is used as the error metric.

$$E^{\nu} = \frac{\left(L^2\right)^{\nu}}{\left(L^2\right)^{\nu=1}} \tag{97}$$

When plotting the logarithm of the error at a given iterate, $\nu + 1$, versus the logarithm of the error at the previous iterate, ν, the function should be linear with a slope of 2 and intercept $\log\left(a\right)$.

$$\frac{\partial\left[\log E^{\nu+1}\right]}{\partial[\log E^{\nu}]} = 2 \tag{98}$$

It is evident that no point will exist for the final (converged) iteration in Newton's method since the error metric is zero. It is also important to note that demonstrating second-order nonlinear convergence does not imply that the method is converging to the correct answer for a given problem. Similarly, the absence of second-order convergence does not imply that the solution is incorrect. Second-order nonlinear convergence simply proves that the Jacobian matrix contains the correct values for the implemented residual equations, and the efficiency of the solution procedure is maximized. Incorrect residual equations can still exhibit second-order convergence if the associated sensitivities were correctly derived and implemented.

The verification problems outlined in this section are a small subset of a large suite of problems. They present the general strategy for verification that has been adopted in this code development process. In general, a verification problem exists for each boundary condition, and each exercises several terms in the governing equations. Currently, there is no verification problem that exercises all nonlinear terms in the coupled governing equations at the same time, but this is the subject of future work. The verification problems also serve as the regression test suite. Verification and regression tests are performed across multiple architectures including compilers and MPI libraries. Finally, all solutions presented in this study were solved in parallel, but the verification and regression tests are also performed in serial.

III.A. Transient Conduction with Variable Properties

III.A.1. Problem Statement

This problem has been previously used to verify a thermal response code as outlined by Amar,[15] and it is intended to verify the implementation of the two dimensional nonlinear transient heat conduction terms in the governing equations and their requisite terms for Newton's method. Consider a non-ablating, non-decomposing, two dimensional, uniform density planar slab of equal length and width subject to a constant specified heat flux on one face, $x = 0$, and adiabatic on the other three faces. Since conduction will only be occurring in one dimension, the governing equation, boundary conditions, and initial conditions that characterize the problem are

$$\left.\begin{aligned} \rho C_p \frac{\partial T}{\partial t} &= \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) \text{ for } 0 \le x \le L \\ -k\frac{\partial T}{\partial x}\bigg|_{x=0} &= \dot{q}_{spec} \\ -k\frac{\partial T}{\partial x}\bigg|_{x=L} &= 0 \\ T\left(\boldsymbol{x}, t=0\right) &= T_o \end{aligned}\right\} \tag{99}$$

and the parameters for this problem are given in Table 2. The thermal conductivity and specific heat are given in Table 3 and are assumed to vary linearly with temperature while the thermal diffusivity is constant giving

$$k\left(T\right) = k_1 + \frac{k_2 - k_1}{T_2 - T_1}\left(T - T_1\right) \tag{100}$$

$$C_p\left(T\right) = C_{p,1} + \frac{C_{p,2} - C_{p,1}}{T_2 - T_1}\left(T - T_1\right) \tag{101}$$

$$\alpha = \frac{k_1}{\rho C_{p,1}} = \frac{k_2}{\rho C_{p,2}} = \frac{k\left(T\right)}{\rho C_p\left(T\right)} \tag{102}$$

Table 2. Variable properties problem parameters.

$T_o = 300\,\mathrm{K}$
$\dot{q}_{spec} = 7.5 \times 10^5\ \mathrm{W/m^2}$
$L = 0.01\,\mathrm{m}$
$\rho = 8000\ \mathrm{kg/m^3}$
$\alpha = k/(\rho C_p) = 2.5 \times 10^{-6}\ \mathrm{m^2/s}$

Table 3. Material properties for temperature dependent properties verification problem.

T, K	k, $\mathrm{W/m{\cdot}K}$	C_p, $\mathrm{J/kg{\cdot}K}$
$T_1 = 300$	10	500
$T_2 = 1300$	100	5000

III.A.2. Analytic Solution

The analytical solution to this problem can be found by first applying Kirchhoff's transformation as described by Arpaci[34] and then using the related constant property solution shown in Beck et al.[35]. The transformed energy equation now has the same form as the related constant property case, and the analytical solution is

$$\frac{\theta\left(x,t\right)-\theta_o}{\dot{q}_{spec}L/k_1}=\frac{\alpha t}{L^2}+\frac{1}{3}-\frac{x}{L}+\frac{1}{2}\left(\frac{x}{L}\right)^2-\frac{2}{\pi^2}\sum_{n=1}^{\infty}\frac{1}{n^2}\exp\left(-n^2\pi^2\frac{\alpha t}{L^2}\right)\cos\left(n\pi\frac{x}{L}\right) \tag{103}$$

where

$$\theta=\left(T-T_1\right)+\frac{k_2-k_1}{T_2-T_1}\frac{1}{2k_1}\left(T-T_1\right)^2 \tag{104}$$

III.A.3. Grid Refinement and Nonlinear Convergence Studies

Although the solution is governed by a one dimensional differential equation, the problem can be used to verify the two dimensional heat conduction capability in the code. Typically, when verifying two or three dimensional codes, the developer would like to exercise flux terms in all directions. This is especially important when dealing with solvers that only use structured meshes since fluxes in the i, j, and k directions may be evaluated with different routines. For an unstructured code, fluxes in all direction are not necessary. This is especially evident when dealing with triangular elements since no faces on the interior of the domain are aligned with the flux direction. The problem was solved on a series of four quadrilateral and four triangle meshes as outlined in Tables 4 and 5 respectively, and the coarse meshes can be seen in Figure 2. Each mesh was uniformly refined by a factor of two to reach the next grid level.

Table 4. Quadrilateral mesh parameters for variable property conduction problem.

Grid Name	N	N_e	$\sqrt{N_e}$	Δt, sec
Coarse	441	400	20	4
Medium	1681	1600	40	2
Fine	6561	6400	80	1
Very Fine	25,921	25,600	160	0.5

Table 5. Triangle mesh parameters for variable property conduction problem.

Grid Name	N	N_e	$\sqrt{N_e}$	Δt, sec
Coarse	787	1492	38.6	4
Medium	2980	5798	76.1	2
Fine	11,757	23,192	152	1
Very Fine	44,322	88,002	297	0.5

The grid convergence results can be seen in Figure 3. For this problem, the normalized L^2 of the temperature solution error vector is used as the error metric where $Q_o = T_o = 300\,\mathrm{K}$ in Eq. (92). It is evident that both element types exhibit second order spatial and temporal accuracy. Also as expected, the quadrilateral elements provide a significant accuracy advantage over triangles.

Since this problem is nonlinear, it is also necessary to verify the convergence rate of the Newton scheme. Figure 4 shows the nonlinear convergence results for the first time step on the coarse quadrilateral mesh. This result is representative of the convergence history throughout the problem for any of the meshes in this study.

III.B. Three Dimensional Transient Conduction with Perfect Gas Convective Boundary Conditions

III.B.1. Problem Statement

Consider a Cartesian aligned uniform density constant property cube whose center is at the origin and has side length, $2L$, that is exposed to a convective heat flux with constant recovery temperature on all faces, but the heat transfer

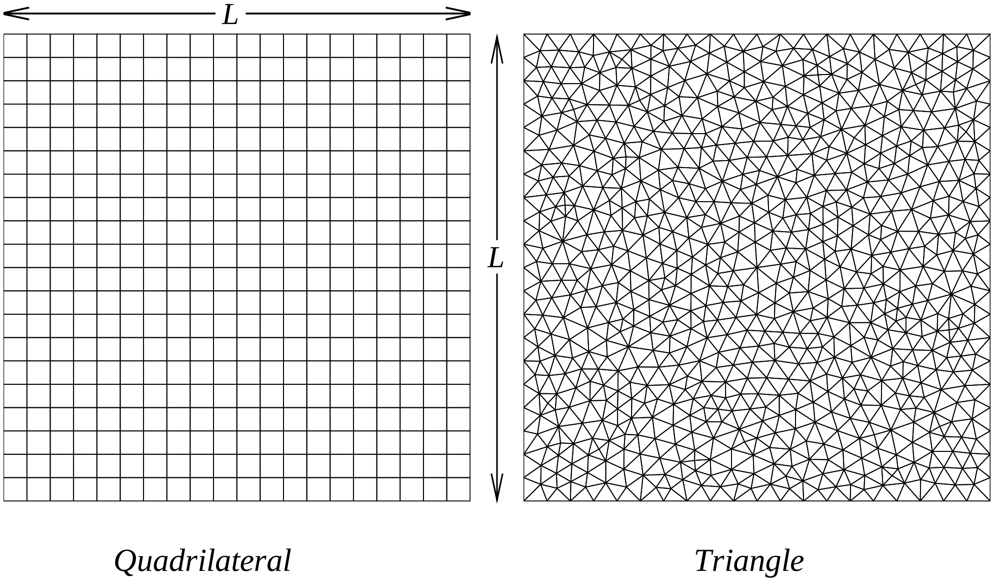


Figure 2. Coarse two dimensional meshes.

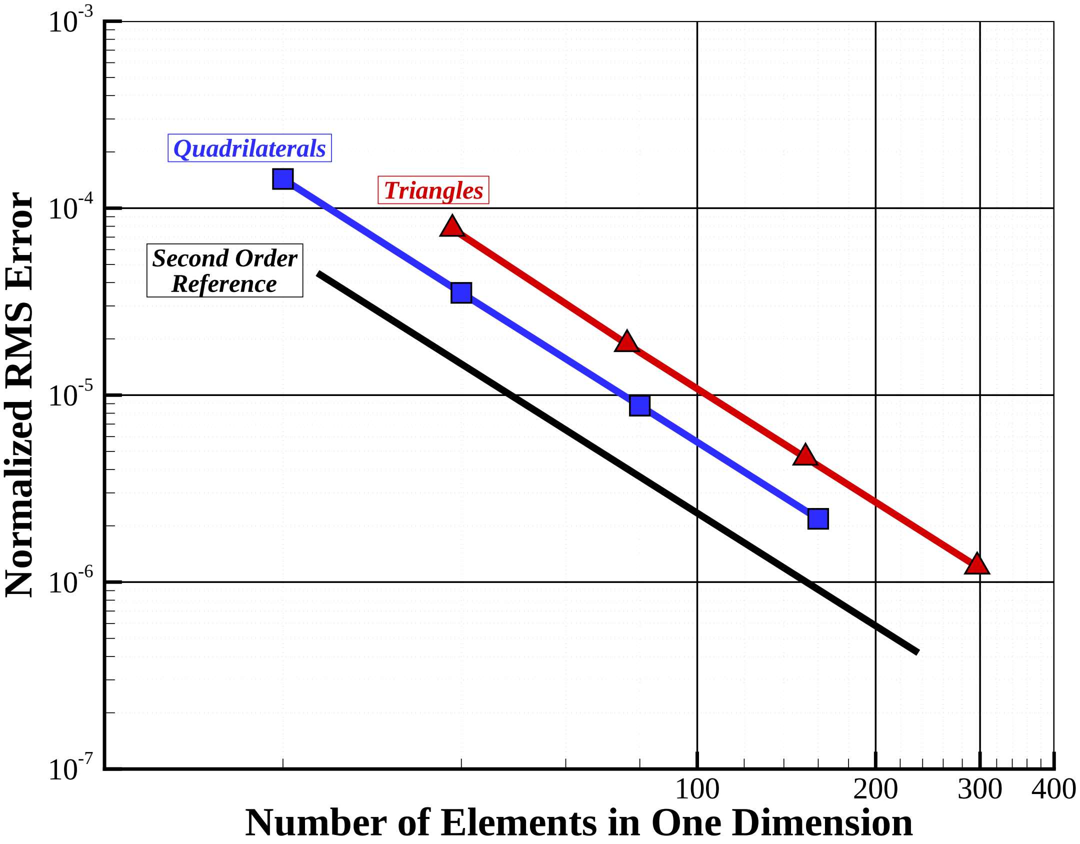


Figure 3. Grid convergence results: variable property conduction problem.

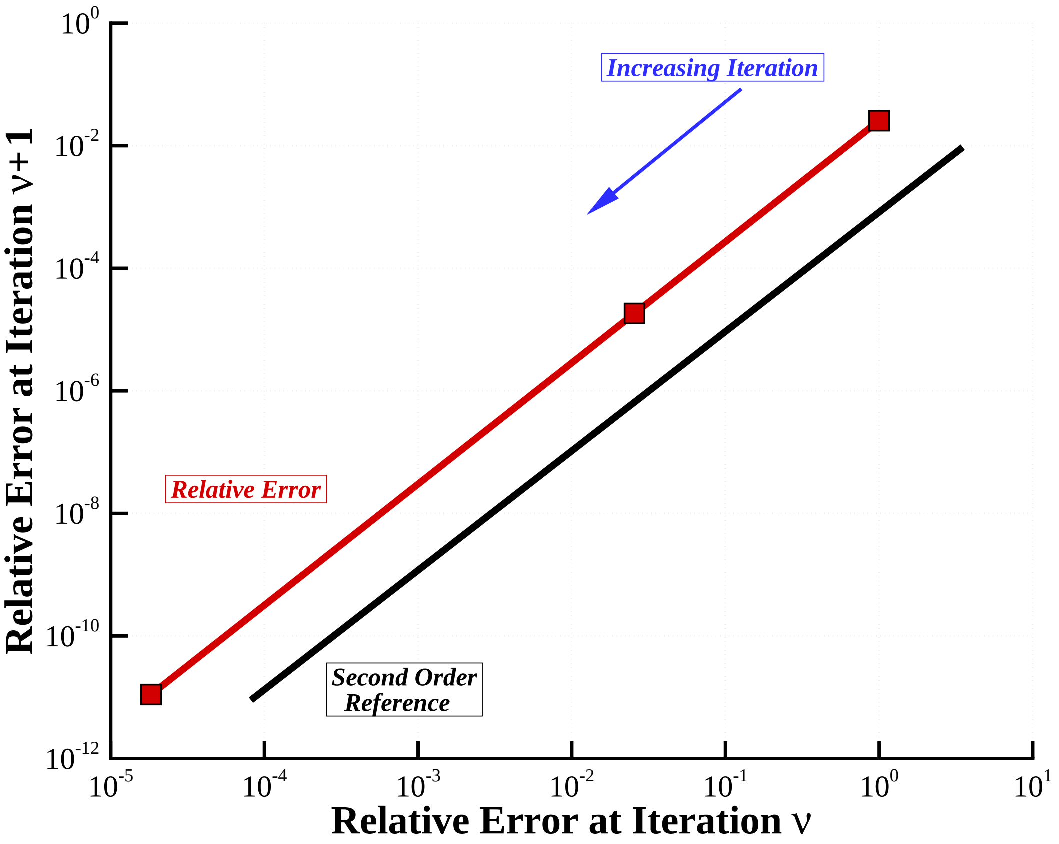


Figure 4. Nonlinear convergence results: variable property conduction problem.

coefficients are only the same on opposing faces of the cube. The governing equation, boundary conditions, and initial conditions that characterize the problem are

$$\left.\begin{aligned} \rho C_p \frac{\partial T}{\partial t} - \boldsymbol{\nabla}\cdot k \boldsymbol{\nabla} T = 0 \\ k\boldsymbol{\nabla}T \cdot \hat{\boldsymbol{n}} = H_x \left(T_r - T\right) \text{ for } x = -L \text{ and } x = L \text{ faces} \\ k\boldsymbol{\nabla}T \cdot \hat{\boldsymbol{n}} = H_y \left(T_r - T\right) \text{ for } y = -L \text{ and } y = L \text{ faces} \\ k\boldsymbol{\nabla}T \cdot \hat{\boldsymbol{n}} = H_z \left(T_r - T\right) \text{ for } z = -L \text{ and } z = L \text{ faces} \\ T\left(\boldsymbol{x}, t = 0\right) = T_o \end{aligned}\right\} \tag{105}$$

and the parameters for this problem are given in Table 6.

Table 6. Cube convection problem parameters.

$T_o = 300\,\mathrm{K}$
$T_r = 1300\,\mathrm{K}$
$L = 0.005\,\mathrm{m}$
$\rho = 8000\,{}^{\mathrm{kg}}/_{\mathrm{m}^3}$
$C_p = 500\,{}^{\mathrm{J}}/_{\mathrm{kg}\cdot\mathrm{K}}$
$k = 10\,{}^{\mathrm{W}}/_{\mathrm{m}\cdot\mathrm{K}}$
$\alpha = k/(\rho C_p) = 2.5 \times 10^{-6}\,{}^{\mathrm{m}^2}/_{\mathrm{s}}$
$H_x = 1500\,{}^{\mathrm{W}}/_{\mathrm{m}^2\mathrm{K}}$
$H_y = 1000\,{}^{\mathrm{W}}/_{\mathrm{m}^2\mathrm{K}}$
$H_z = 500\,{}^{\mathrm{W}}/_{\mathrm{m}^2\mathrm{K}}$

III.B.2. Analytic Solution

One dimensional solutions to the heat equation with constant thermal conductivity and specific heat can be combined to give exact solutions to simple two and three dimensional geometries as outlined in Bejan.[36] For this problem, the exact solution used is for the one dimensional constant property planar slab of length L with front face convective environment and an adiabatic back face as outlined in Arpaci.[34] The exact solution in the x-direction is given by

$$\theta_x\left(x,t\right) = \frac{T(x,t) - T_r}{T_o - T_r} = 2\sum_{n=1}^{\infty}\left(\frac{\sin\nu_n}{\nu_n + \sin\nu_n\cos\nu_n}\right)\exp\left(-\nu_n^2\alpha t/L^2\right)\cos\left(\nu_n x/L\right) \tag{106}$$

for all $\nu_n > 0$ where the eigenvalues can be found by solving the implicit relationship

$$\nu_n \sin\nu_n = \mathrm{Bi}\cos\nu_n \tag{107}$$

and the Biot number is given by

$$\mathrm{Bi} = \frac{H_x L}{k} \tag{108}$$

and there is an analogous expression for $\theta_y\left(y,t\right)$ and $\theta_z\left(z,t\right)$. The one dimensional solutions can be combined to give the three dimensional distribution according to

$$\frac{T(\boldsymbol{x},t) - T_r}{T_o - T_r} = \theta_x\theta_y\theta_z \tag{109}$$

The above expression is only valid for the octant where $x \geq 0$, $y \geq 0$, and $z \geq 0$. Since the solution is symmetric across the $x = 0$, $y = 0$, and $z = 0$ planes, Eq. (109) can be appropriately applied to give the temperature distribution in the entire domain. The solution after 20 seconds on the cube's exposed faces can be seen in Figure 5.

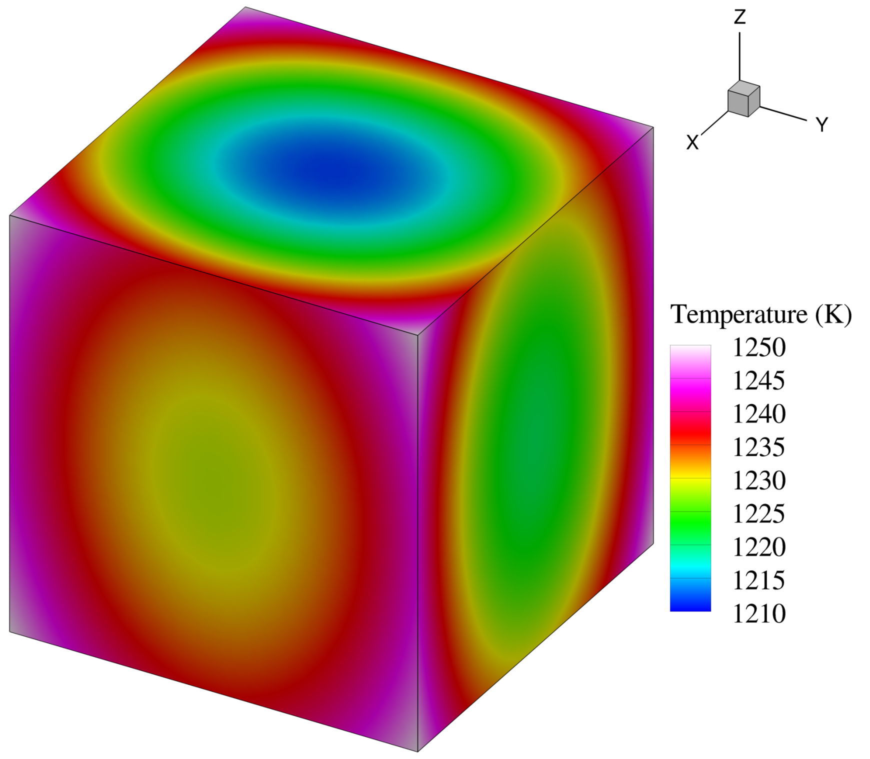


Figure 5. Solution at 20 seconds for cube convection problem.

III.B.3. Grid Refinement Study

The problem was solved on a series of three hexahedron, three prism, and three tetrahedron meshes as outlined in Tables 7, 8, and 9 respectively, and the coarse meshes can be seen in Figure 6. For this problem, the normalized L^2 of the temperature solution error vector is used as the error metric where $Q_o = T_o = 300\,\mathrm{K}$ in Eq. (92). The results of the grid refinement study can be seen in Figure 7, and it is evident that the code exhibits both second order spatial and temporal accuracy. As expected, the hexahedrons were the most accurate for a given mesh size while the tetrahedrons were the least accurate. Although the boundary condition is nonlinear, the interior solution is linear resulting in a very weakly nonlinear problem. Consequently, the Newton scheme converged in too few iterations to generate a meaningful trend for a nonlinear convergence study.

Table 7. Hexahedron mesh parameters for cube convection problem.

Grid Name	N	N_e	$\sqrt[3]{N_e}$	Δt, sec
Coarse	9261	8000	20	0.5
Medium	68,921	64,000	40	0.25
Fine	531,441	512,000	80	0.125

Table 8. Prism mesh parameters for cube convection problem.

Grid Name	N	N_e	$\sqrt[3]{N_e}$	Δt, sec
Coarse	16,275	29,360	30.8	0.5
Medium	122,754	233,040	61.5	0.25
Fine	956,286	1,863,200	123	0.125

Table 9. Tetrahedron mesh parameters for cube convection problem.

Grid Name	N	N_e	$\sqrt[3]{N_e}$	Δt, sec
Coarse	19,414	103,268	46.9	0.5
Medium	130,932	736,834	90.3	0.25
Fine	957,152	5,571,020	177	0.125

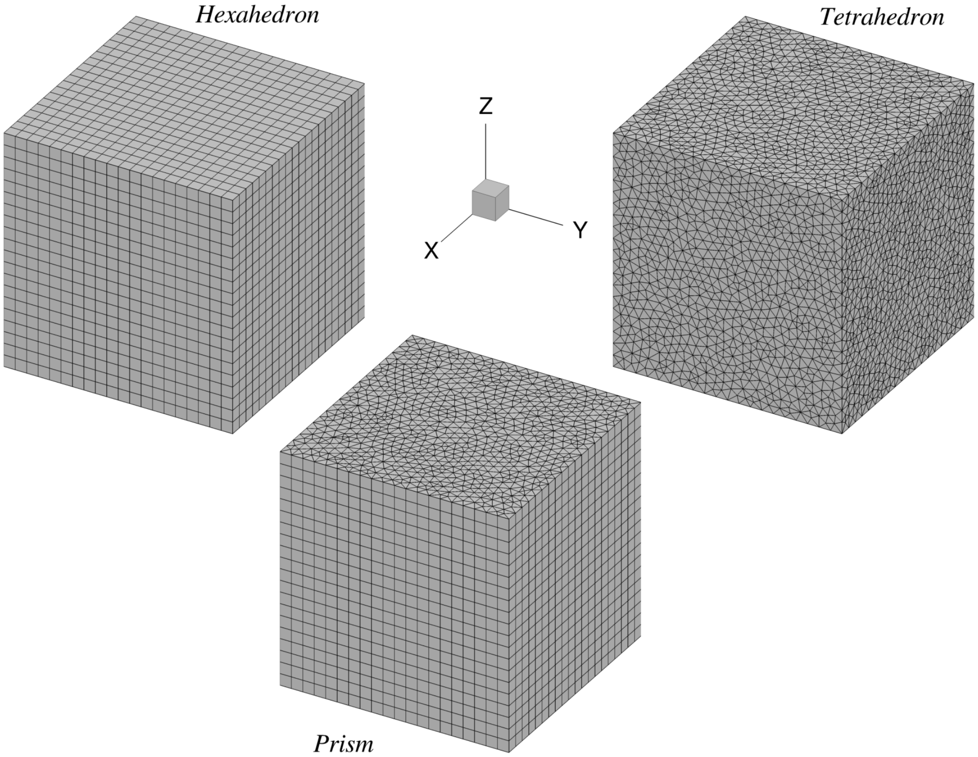


Figure 6. Coarse cube meshes.

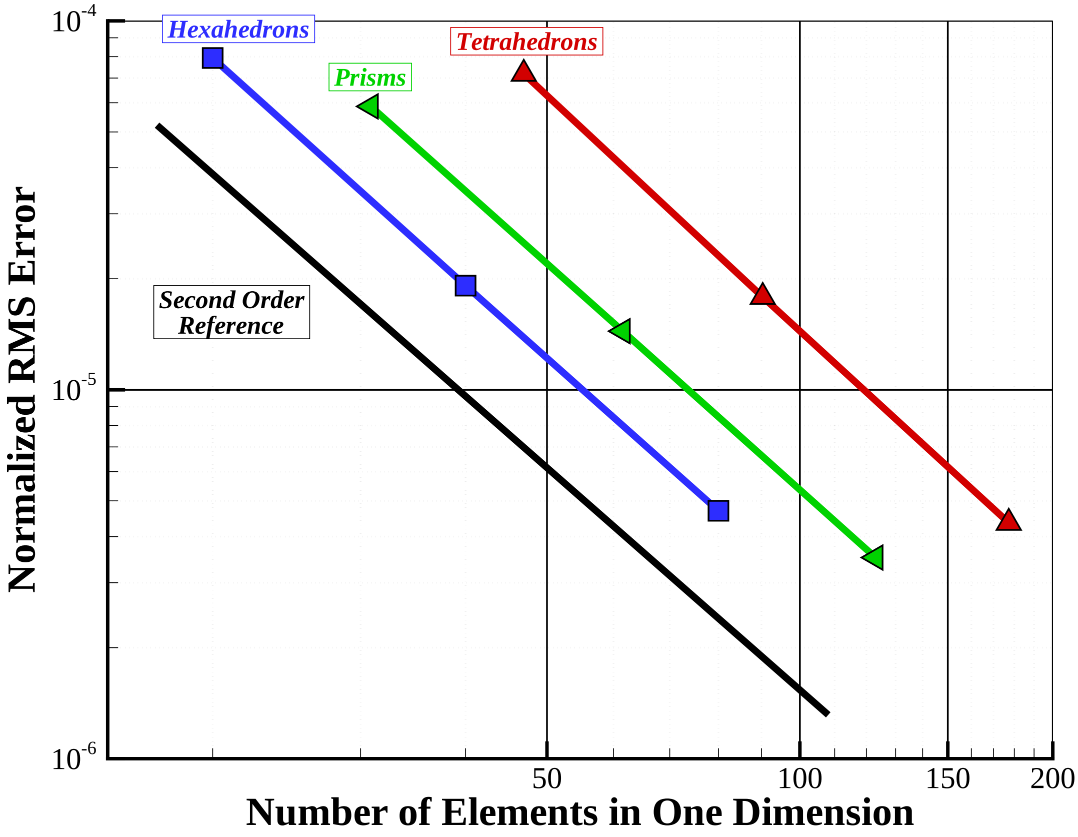


Figure 7. Grid convergence results: cube convection problem.

III.C. Three Dimensional Transient Conduction with Manufactured Solution

Often it is difficult to find an analytic solution that exercises more than a few terms in the governing equations of interest. It is desirable to have an exact solution to a problem that exercises not only the nonlinear terms in the governing equations but also several code options simultaneously. One common approach for developing such a verification problem is the Method of Manufactured Solutions.[20,22] For a general PDE with temperature as the independent variable

$$\mathcal{D}\left(T\right) = 0 \tag{110}$$

where $\mathcal{D}$ represents the differential operator, a simplified outline to the approach is as follows:

1. Choose the desired solution, $T\left(\boldsymbol{x}, t\right)$, to the PDE.
   - Solution should be composed of simple analytic functions that are functions of both space and time if desired.
   - Solution should be sufficiently smooth so that the theoretical order of accuracy can be achieved.
   - Solution should exercise all desired terms in the PDE.
   - Solution should be differentiable at least as many times that is dictated by the PDE.
   - Solution should not violate physics such that the software will not run. For example, negative densities or temperatures on an absolute scale should not be part of the solution.
   - Like all verification problems, the solution should not be limited to something that is physically meaningful or of practical importance since verification is simply a mathematical exercise.
2. Choose analytic model equations for any properties that depend on the independent variable (e.g. thermal conductivity and specific heat for the energy equation).
3. Substitute the solution and property models into the PDE, and allow the differential operator, $\mathcal{D}$, to operate on these functions.
4. In general, the chosen solution is not a solution to the PDE, $\mathcal{D}\left(T\right) \neq 0$. However, the PDE can be redefined to have a source term.
   $$\mathcal{D}\left(T\right) = \dot{Q}\left(\boldsymbol{x}, t\right) \tag{111}$$
   where $\dot{Q}\left(\boldsymbol{x}, t\right)$ is the source term which is defined by the result of step 3.
5. Implement the resulting source term function in the software since T is the exact solution to the modified PDE in Eq. (111) with known source term.
6. Perform verification studies to check for orders of accuracy and convergence rate

III.C.1. Solution Development

The transient temperature distribution, $T\left(\boldsymbol{x}, t\right)$, in a constant density material is governed by

$$\rho C_p \frac{\partial T}{\partial t} - \boldsymbol{\nabla}\cdot(k\boldsymbol{\nabla}T) = \dot{Q} \tag{112}$$

and the manufactured temperature solution is chosen to be a function of cosines

$$T(\boldsymbol{x}, t) = \theta\cos(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) + T_o \tag{113}$$

where T_o is the average temperature about which the solution oscillates with amplitude θ. The models for thermal conductivity and specific heat are chosen to be quadratic in temperature.

$$k\left(T\right) = k_0 + k_1\left(\frac{T - T_o}{\theta}\right) + k_2\left(\frac{T - T_o}{\theta}\right)^2 \tag{114}$$

$$C_p\left(T\right) = C_{p_0} + C_{p_1}\left(\frac{T - T_o}{\theta}\right) + C_{p_2}\left(\frac{T - T_o}{\theta}\right)^2 \tag{115}$$

Substituting Eqs. (113) (114) and (115) into Eq. (112) and solving for the source term gives

$$
\begin{aligned}
\dot{Q}\left(\boldsymbol{x},t\right) =&\rho\,[\; C_{p_0}+C_{p_1}\cos(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&+C_{p_2}\cos^2(A_x x+A_t t)\cos^2(B_y y+B_t t)\cos^2(C_z z+C_t t)\cos^2(D_t t)\;]\\
&\{\;-\theta\,[\;A_t\sin(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&+B_t\cos(A_x x+A_t t)\sin(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&+C_t\cos(A_x x+A_t t)\cos(B_y y+B_t t)\sin(C_z z+C_t t)\cos(D_t t)\\
&+D_t\cos(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\sin(D_t t)\;]\;\}\\
&-\{\;[\;k_0+k_1\cos(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&+k_2\cos^2(A_x x+A_t t)\cos^2(B_y y+B_t t)\cos^2(C_z z+C_t t)\cos^2(D_t t)\;]\\
&\left[-\theta\left(A_x^2+B_y^2+C_z^2\right)\cos(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\right]\\
&-\theta\,[\;A_x\sin(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&(\;-k_1A_x\sin(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&+2k_2A_x\sin(A_x x+A_t t)\cos(A_x x+A_t t)\cos^2(B_y y+B_t t)\cos^2(C_z z+C_t t)\cos^2(D_t t)\;)\\
&+B_y\cos(A_x x+A_t t)\sin(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&(\;-k_1B_y\cos(A_x x+A_t t)\sin(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&+2k_2B_y\cos^2(A_x x+A_t t)\sin(B_y y+B_t t)\cos(B_y y+B_t t)\cos^2(C_z z+C_t t)\cos^2(D_t t)\;)\\
&+C_z\cos(A_x x+A_t t)\cos(B_y y+B_t t)\sin(C_z z+C_t t)\cos(D_t t)\\
&(\;-k_1C_z\cos(A_x x+A_t t)\cos(B_y y+B_t t)\sin(C_z z+C_t t)\cos(D_t t)\\
&+2k_2C_z\cos^2(A_x x+A_t t)\cos^2(B_y y+B_t t)\sin(C_z z+C_t t)\cos(C_z z+C_t t)\cos^2(D_t t)\;)\;\}
\end{aligned}
\tag{116}
$$

Notice that the source term turns out to be a complicated function which is typical of MMS. The source terms can be derived by hand, but a symbolic manipulator such as Maple[37] or Mathematica[38] can be very helpful with the derivation. Note that this manufactured solution can be used for several verification exercises including

- One Dimensional Conduction $(B_y = C_z = 0)$
- Two Dimensional Conduction $(C_z = 0)$
- Steady Conduction $(A_t = B_t = C_t = D_t = 0)$
- Linear $(C_{p_1} = C_{p_2} = k_1 = k_2 = 0)$

The problem parameters used in this study are given in Table 10.

Both specified temperature and specified heat flux boundary conditions can be exercised with this problem. For the specified temperature condition, the time and quadrature point coordinates are simply plugged into Eq. (113) to give the boundary condition. The boundary heat flux at any point can be determined according to

$$
\begin{aligned}
\dot{q}_{cond_s}=-k\boldsymbol{\nabla}T\cdot\hat{\boldsymbol{n}}=&-\,[k_0+k_1\cos(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)\\
&+k_2\cos^2(A_x x+A_t t)\cos^2(B_y y+B_t t)\cos^2(C_z z+C_t t)\cos^2(D_t t)]\\
&\{-\theta\,[A\sin(A_x x+A_t t)\cos(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)n_x\\
&+B\cos(A_x x+A_t t)\sin(B_y y+B_t t)\cos(C_z z+C_t t)\cos(D_t t)n_y\\
&+C\cos(A_x x+A_t t)\cos(B_y y+B_t t)\sin(C_z z+C_t t)\cos(D_t t)n_z]\}
\end{aligned}
\tag{117}
$$

where

$$
\hat{\boldsymbol{n}}=n_x\boldsymbol{i}+n_y\boldsymbol{j}+n_z\boldsymbol{k}
\tag{118}
$$

III.C.2. Grid Refinement and Nonlinear Convergence Studies

An arbitrary geometry was chosen for this problem such that not all the faces are Cartesian aligned nor are they all flat. The problem was solved on a series of three hexahedron meshes and three tetrahedron meshes as outlined in Tables 11 and 12 respectively, and the coarse meshes can be seen in Figure 8. The initial condition and solution to this problem

Table 10. MMS energy problem parameters.

$T_o = 300\,\mathrm{K}$
$\theta = 60\,\mathrm{K}$
$\rho = 800\,\mathrm{kg}/\mathrm{m}^3$
$C_{p_0} = 500\,\mathrm{J}/\mathrm{kg}\cdot\mathrm{K}$
$C_{p_1} = 50\,\mathrm{J}/\mathrm{kg}\cdot\mathrm{K}$
$C_{p_2} = 5\,\mathrm{J}/\mathrm{kg}\cdot\mathrm{K}$
$k_0 = 30\,\mathrm{W}/\mathrm{m}\cdot\mathrm{K}$
$k_1 = 3\,\mathrm{W}/\mathrm{m}\cdot\mathrm{K}$
$k_2 = 0.3\,\mathrm{W}/\mathrm{m}\cdot\mathrm{K}$
$A_x = B_y = C_z = 1\,\mathrm{m}^{-1}$
$A_t = B_t = C_t = 0.1\,\mathrm{sec}^{-1}$
$D_t = 0.05\,\mathrm{sec}^{-1}$

Table 11. Hexahedron mesh parameters for MMS problems.

Grid Name	N	N_e	$\sqrt[3]{N_e}$	Δt, sec
Coarse	18,081	16,000	25.1	1.0
Medium	136,161	128,000	50.4	0.5
Fine	1,056,321	1,024,000	101	0.25

Table 12. Tetrahedron mesh parameters for MMS problems.

Grid Name	N	N_e	$\sqrt[3]{N_e}$	Δt, sec
Coarse	20,657	107,842	47.6	1.0
Medium	133,029	736,709	90.3	0.5
Fine	933,416	5,375,076	175	0.25

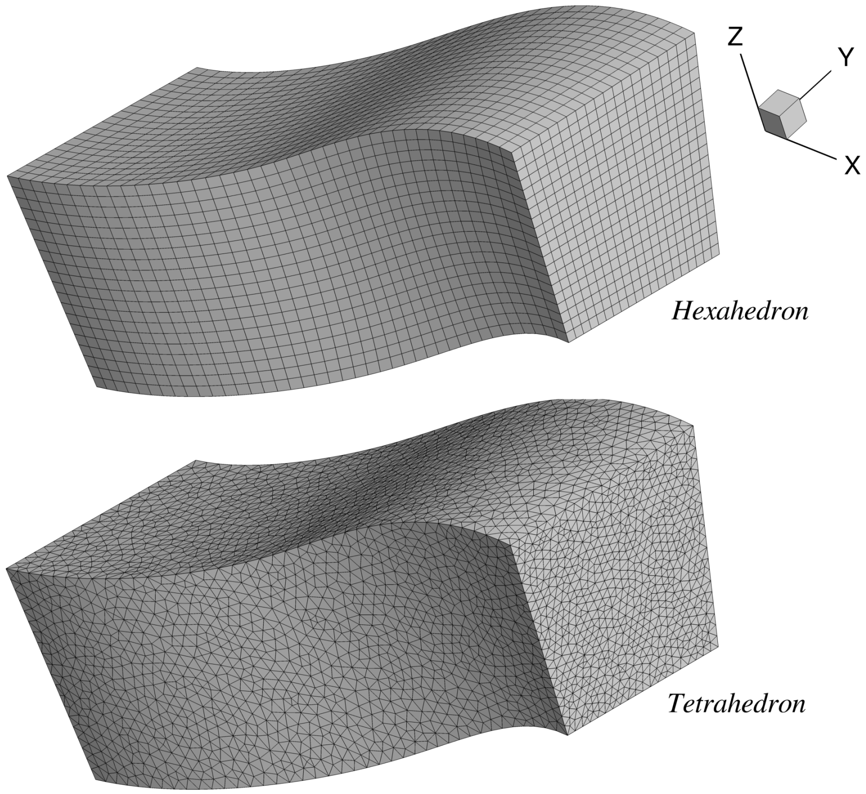


Figure 8. Coarse grids for MMS problems.

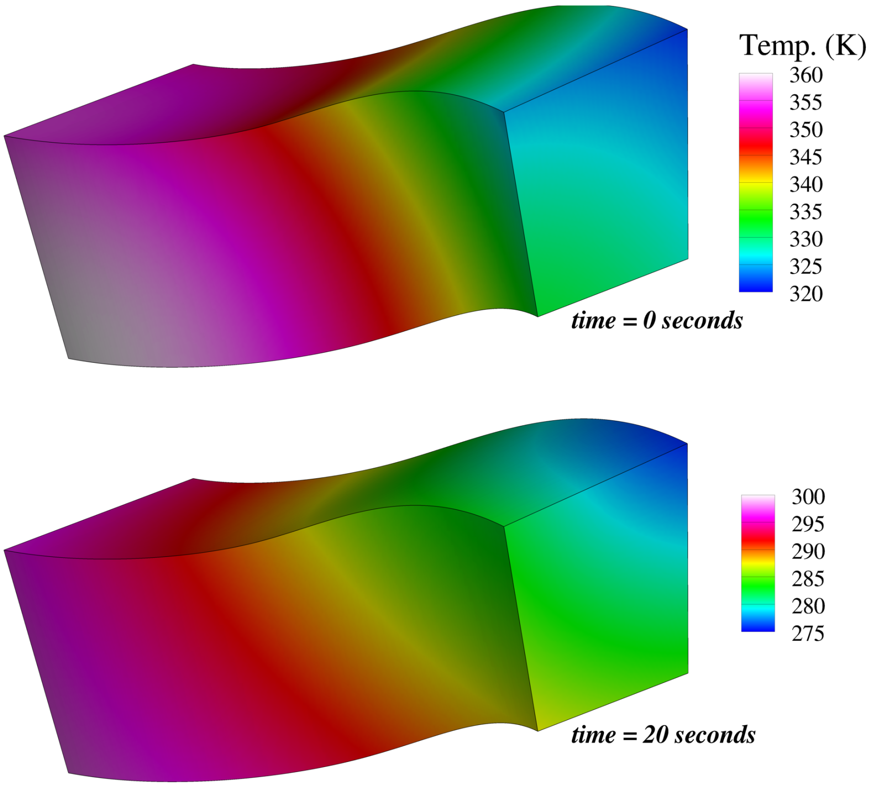


Figure 9. Solution for the MMS energy problem.

can be seen in Figure 9. The problem was solved with a specified flux on the four long sides while the two smaller sides had a specified temperature boundary condition.

For this problem, the normalized L^2 of the temperature solution error vector is used as the error metric where $Q_o = T_o = 300\,\mathrm{K}$ in Eq. (92). The results of the verification study can be seen in Figure 10, and it is evident that the code exhibits both second order spatial and temporal accuracy. The nonlinear convergence study confirms that the Newton scheme is exhibiting second order convergence as seen in Figure 11.

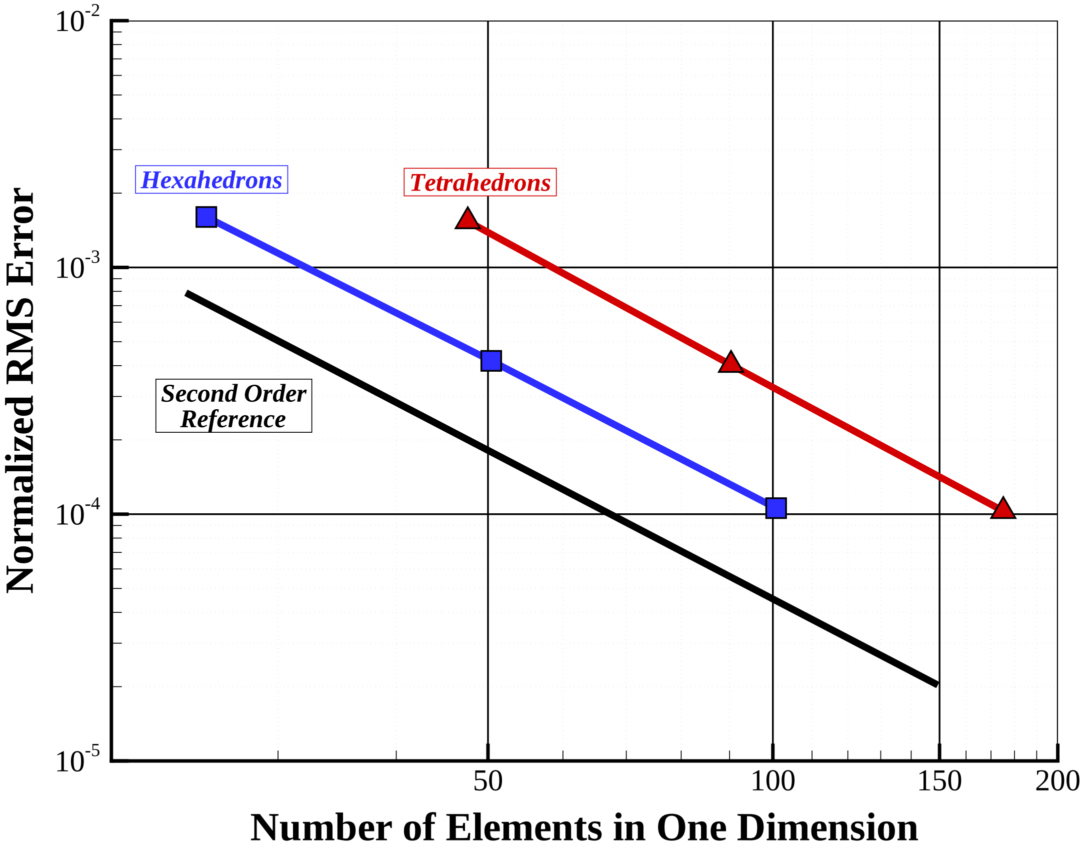


Figure 10. Grid convergence results: MMS energy problem.

III.D. Three Dimensional Transient Porous Gas Flow with Manufactured Solution

III.D.1. Solution Development

The transient gas density distribution in a porous medium with Darcian flow is governed by

$$\frac{\partial(\phi\rho_g)}{\partial t} - \boldsymbol{\nabla}\cdot\left(\rho_g \frac{\widetilde{\boldsymbol{\kappa}}}{\mu}\boldsymbol{\nabla}P\right) = \dot{\omega}_g \tag{119}$$

If it is further assumed that the medium is isothermal, it has uniform isotropic permeability and constant porosity, and the gas has constant viscosity and behaves according to the perfect gas law, then the governing equation simplifies to

$$\phi\frac{\partial\rho_g}{\partial t} - \frac{\kappa RT}{\mu}\boldsymbol{\nabla}\cdot(\rho_g\boldsymbol{\nabla}\rho_g) = \dot{\omega}_g \tag{120}$$

The manufactured density solution is chosen to be a function of cosines

$$\rho_g(\boldsymbol{x},t) = \theta\cos(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) + \rho_{g_o} \tag{121}$$

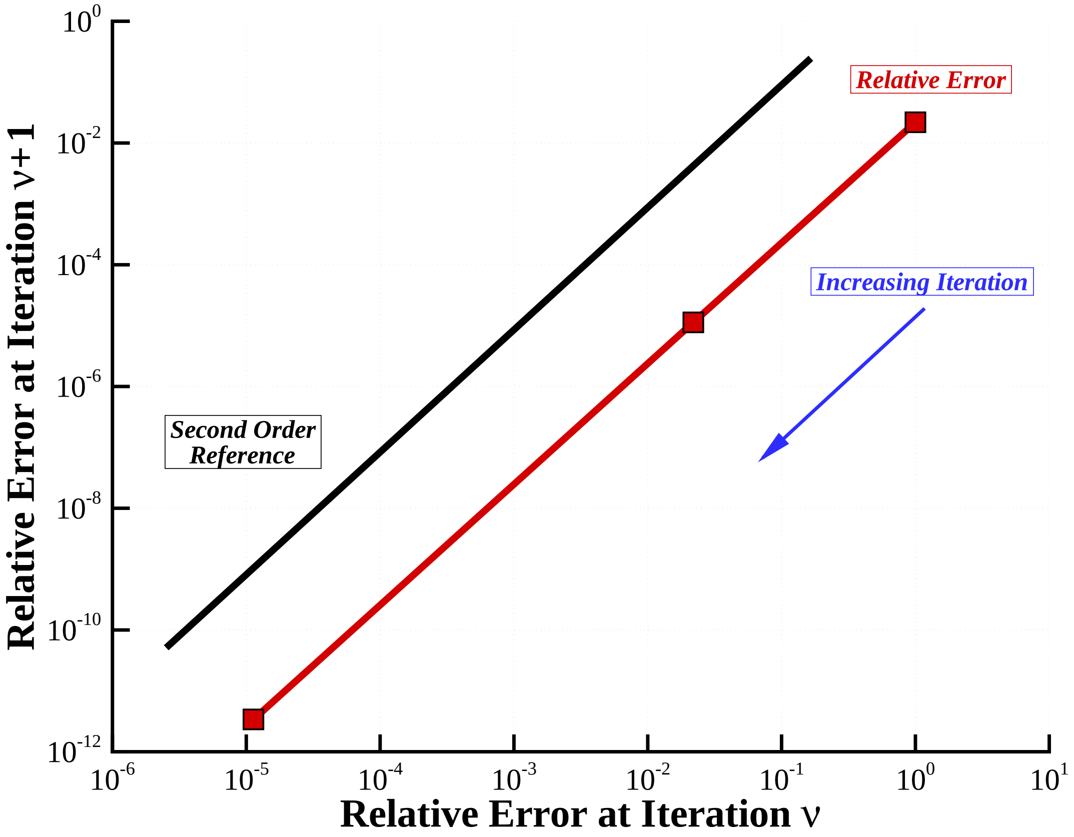


Figure 11. Nonlinear convergence results: MMS energy problem.

where ρ_{g_o} is the average density about which the solution oscillates with amplitude θ. The source term is found by substituting the manufactured solution, Eq. (121) , into the governing equation, Eq. (120), which gives

$$
\begin{aligned}
\dot{\omega}_g(\boldsymbol{x},t) = -\phi\theta\,[\,& A_t \sin(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) \\
&+ B_t \cos(A_x x + A_t t)\sin(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) \\
&+ C_t \cos(A_x x + A_t t)\cos(B_y y + B_t t)\sin(C_z z + C_t t)\cos(D_t t) \\
&+ D_t \cos(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\sin(D_t t)\ ] \\
-\frac{\kappa RT}{\mu}\,\{ & -\rho_{g_o}\theta\,(\,A_x^2 + B_y^2 + C_z^2\,)\cos(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) \\
& -\theta^2\,(\,A_x^2 + B_y^2 + C_z^2\,)\cos^2(A_x x + A_t t)\cos^2(B_y y + B_t t)\cos^2(C_z z + C_t t)\cos^2(D_t t) \\
+\theta^2\cos^2(D_t t)\,[\,& A_x^2 \sin^2(A_x x + A_t t)\cos^2(B_y y + B_t t)\cos^2(C_z z + C_t t) \\
&+ B_y^2 \cos^2(A_x x + A_t t)\sin^2(B_y y + B_t t)\cos^2(C_z z + C_t t) \\
&+ C_z^2 \cos^2(A_x x + A_t t)\cos^2(B_y y + B_t t)\sin^2(C_z z + C_t t)\ ]\ \}
\end{aligned}
\tag{122}
$$

Like the MMS energy problem, this solution can also be used for several verification exercises including

- One Dimensional Flow $(B_y = C_z = 0)$
- Two Dimensional Flow $(C_z = 0)$
- Steady Flow $(A_t = B_t = C_t = D_t = 0)$

Unlike the MMS energy problem, this problem can not be made linear due to the $\rho_g \boldsymbol{\nabla} \rho_g$ term. The parameters used in this study can be seen in Table 13.

Table 13. MMS gas flow problem parameters.

$T = 300\,\mathrm{K}$
$R = 300\,{}^{\mathrm{J}}/_{\mathrm{kg \cdot K}}$
$\theta = 0.2\,{}^{\mathrm{kg}}/_{\mathrm{m}^3}$
$\rho_{g_o} = 1\,{}^{\mathrm{kg}}/_{\mathrm{m}^3}$
$\kappa = 1 \times 10^{-15}\,\mathrm{m}^2$
$\phi = 0.2$
$\mu = 1 \times 10^{-6}\,\mathrm{Pa \cdot s}$
$A_x = B_y = C_z = 1\,\mathrm{m}^{-1}$
$A_t = B_t = C_t = 0.1\,\mathrm{sec}^{-1}$
$D_t = 0.05\,\mathrm{sec}^{-1}$

Specified density, specified pressure, and specified mass flux boundary conditions can be exercised with this problem. For the specified density condition, the time and quadrature point coordinates are simply plugged into Eq. (121) and multiplied by the porosity to give the boundary condition. The boundary pressure at any point can be determined by first calculating the density according to Eq. (121) then using the perfect gas law. The mass flux at the boundary can be calculated according to

$$
\begin{aligned}
\dot{m}_{g_s} = \phi\rho_g \boldsymbol{v}_g \cdot \hat{\boldsymbol{n}} = -\frac{\kappa RT}{\mu}\,&[\theta\cos(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) + \rho_{g_o}] \\
&\{-\theta\,[A_x \sin(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) n_x \\
&+ B_y \cos(A_x x + A_t t)\sin(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) n_y \\
&+ C_z \cos(A_x x + A_t t)\cos(B_y y + B_t t)\sin(C_z z + C_t t)\cos(D_t t) n_y\ ]\ \}
\end{aligned}
\tag{123}
$$

III.D.2. Grid Refinement and Nonlinear Convergence Studies

The geometry used for this problems is the same that was used in the MMS energy problem in Section III.C. The hexahedron and tetrahedron grid parameters can be found in Tables 11 and 12 respectively, and the coarse meshes can be seen in Figure 8. The initial condition and solution to this problem can be seen in Figure 12. The problem was solved with mass flux, density, and pressure boundary conditions, each on two sides.

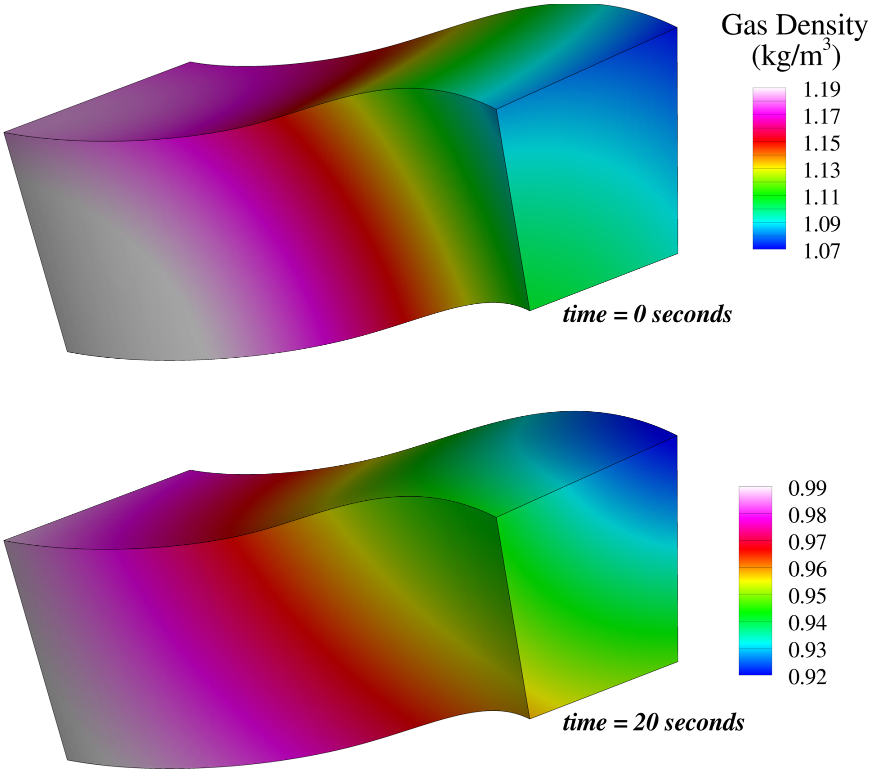


Figure 12. Solution for the MMS gas flow problem.

For this problem, the normalized L^2 of the gas density solution error vector is used as the error metric where $Q_o = \rho_{g_o} = 1.0\,{}^{\mathrm{kg}}\!/\!{}_{\mathrm{m}^3}$ in Eq. (92). The results of the verification study can be seen in Figure 13, and it is evident that the code exhibits both second order spatial and temporal accuracy. The nonlinear convergence study shows that the Newton scheme is exhibiting second order convergence as seen in Figure 14.

IV. Complex-Step Size Study

In Section II.I, the relative perturbation factor, r in Eq. (65), for the complex-step method was discussed. In order to have a Jacobian exact to machine precision, r should be chosen such that underflow occurs on the right-hand-side of Eq. (64). However, this is impossible if the order of magnitude of the Jacobian is not known before hand. In this code development project, $r = 10^{-20}$ is used. If the independent variable is on the order of 10^3, which is typical of temperatures in a re-entry ablation problem, then Eq. (65) says that the step size will be on the order of 10^{-17}, which gives an error term on the order of 10^{-34} in Eq. (64). Consequently, for a calculation with 16 digits of precision (typical for double precision), underflow will occur for Jacobians as small 10^{-18} resulting in exact Jacobians. Once the Jacobian gets smaller it will be approximate, but Jacobians less than 10^{-18} are typically inconsequential to the solution of the linear system. Although $r = 10^{-20}$ is used in this study, values far less than 10^{-20} are acceptable as long as an underflow error is not introduced.

In order to illustrate the effects of the choice of r, one time step of a nonlinear heat conduction problem was solved with various choices of r. The choices range from 10^{-1} to 10^{-100}, and the results can be seen in Figure 15. It is evident that with the larger choices of r, the convergence rate is reduced to first order, but the method still reaches the convergence criteria of 10^{-12}. The optimal second order nonlinear convergence was reached with values of $r \leq 10^{-5}$, and performance remained static with further reductions in r, which is evident from the three co-linear results. Additionally, using a factor of 10^{-100} presented no numerical issues. It is also evident from Figure 15 that the iterations required for convergence is reduced from 10 with $r = 10^{-1}$ to 6 with $r \leq 10^{-5}$ which is directly proportional to the relative computational time for the two cases. From this study, it is evident the the complex-step

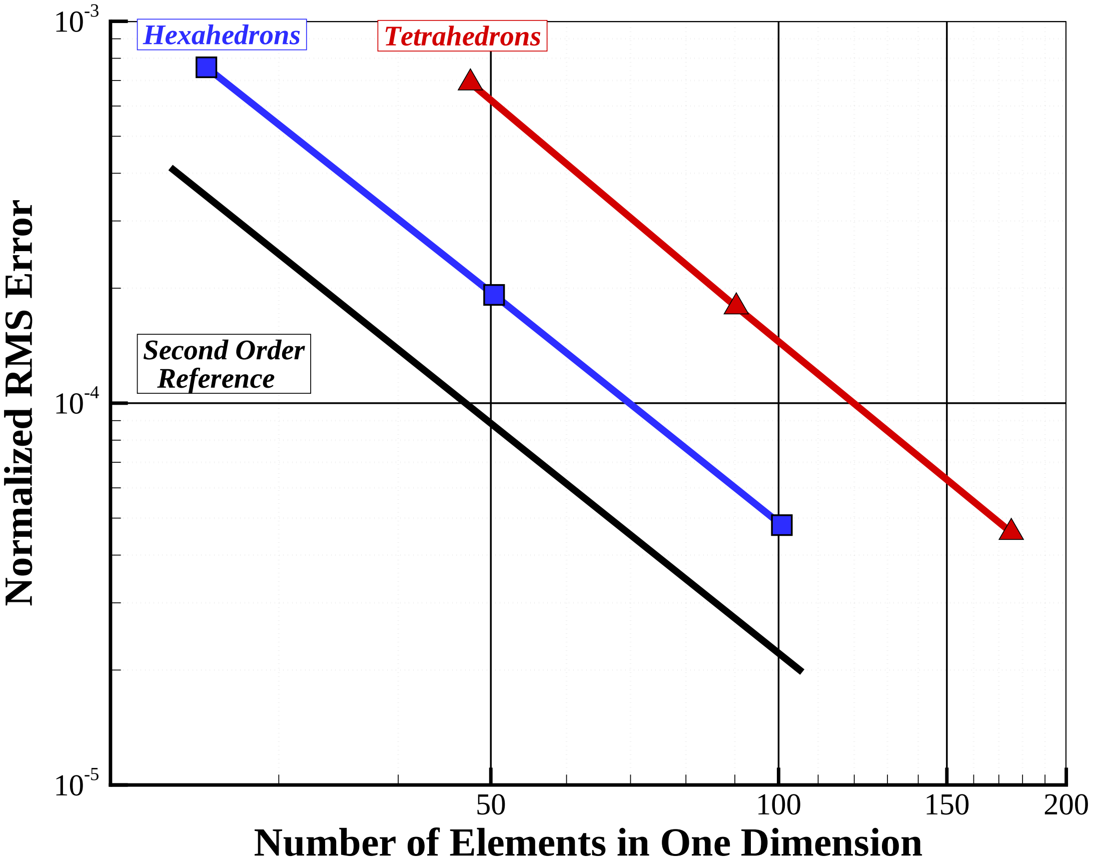


Figure 13. Grid convergence results: MMS gas flow problem.

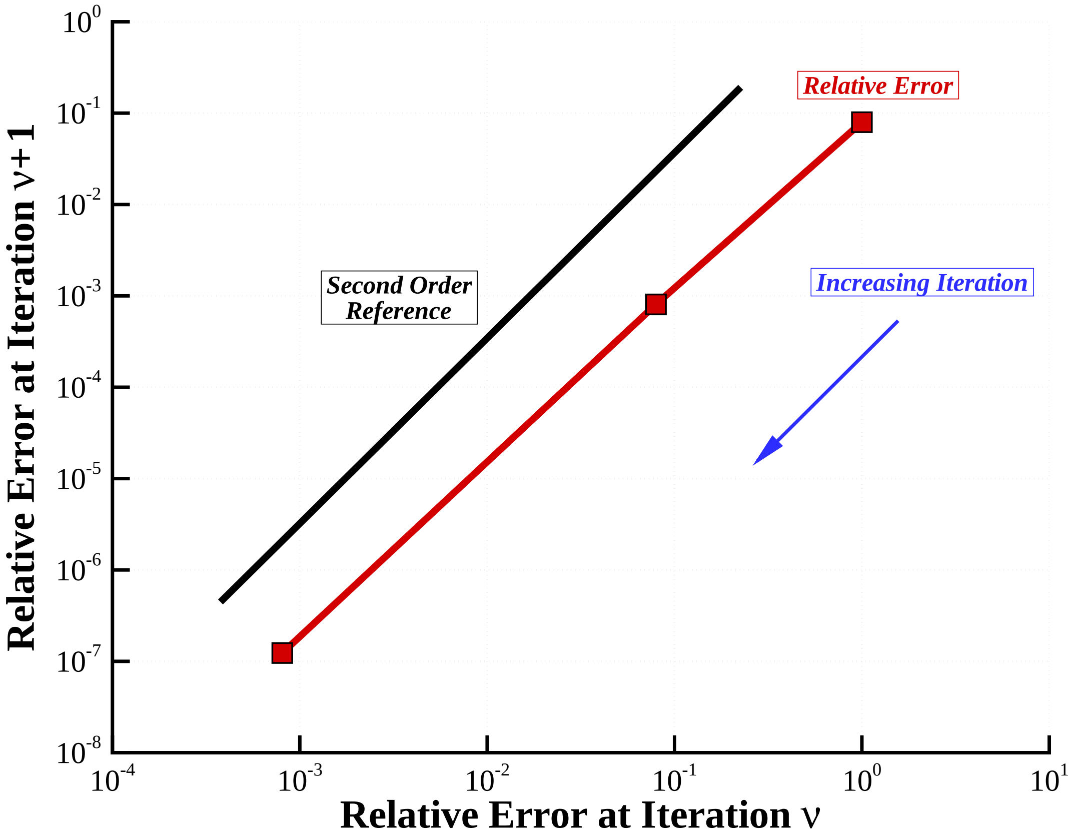


Figure 14. Nonlinear convergence results: MMS gas flow problem.

method is a valid way of calculating Jacobians to machine precision which gives the optimal convergence rate of the Newton scheme. Unlike calculating Jacobians numerically with other methods such as finite-difference, the choice of the perturbation factor for the complex-step method is relatively straightforward.

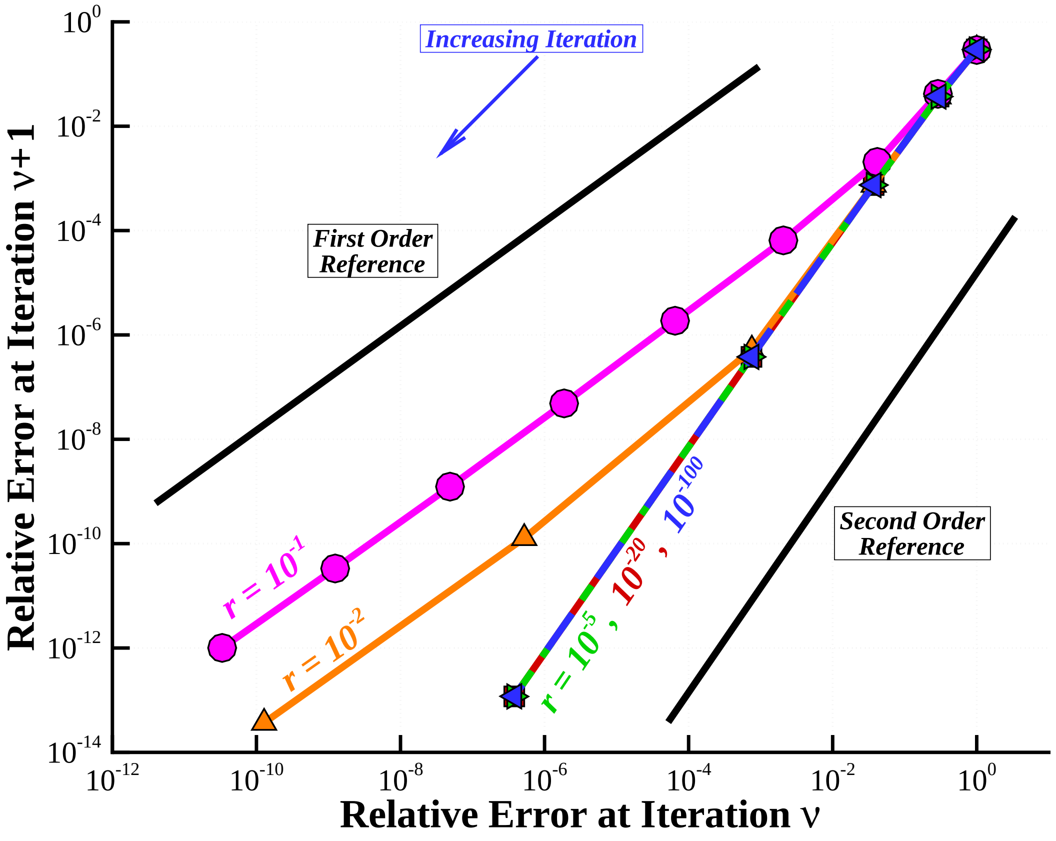


Figure 15. Complex-step size variation.

V. Conclusions and Future Work

Verification results for the Charring Ablating Thermal Protection Implicit System Solver were presented. Several nonlinear terms in the energy and gas continuity equations, as well as many boundary conditions, were verified through comparisons with analytical solutions and MMS. Through the verification process, the code exhibited the expected second order spatial and second order temporal accuracy. New MMS problems were developed for both the energy and gas continuity equations, and their source terms were archived for dissemination to the ablation and heat transfer communities. Newton's method for solving the system of nonlinear equations was verified to exhibit the optimal second order convergence for all problems presented, and the complex-step method was introduced to numerically calculate exact Jacobian terms.

Throughout this work, the verification process helped identify several bugs resulting from both coding and derivation errors. Consequently, the verification exercises have been invaluable in developing confidence in the new software as well as improving robustness, usability, and efficiency.

Future work will include the development of a MMS problem that exercises the coupling terms in the governing equations as well as the decomposition kinetics involved in solving charring ablator problems. No concerted effort has been put into improving the efficiency of the software. Consequently, profiling exercises should be performed. Validation through comparison with experimental data should be done for the models that have been implemented and verified, and the addition of improved physical models, such as ablation models, porous flow laws, and in-depth thermochemical non-equilibrium need to be completed.

VI. Acknowledgments

The authors would like to thank:

- Ben Blackwell for fruitful discussions, mentoring, and most importantly the idea of using the complex-step method to calculate Jacobian terms. This trick has expedited the development process and saved countless headaches.
- Kemelli Estacio-Hiroms whose unpublished work outlines the development of the manufactured solution from which the manufactured solutions in this study were based.
- Brandon Oliver for using the code to solve real problems and providing invaluable feedback.
- Chuck Campbell, Amy Cassady, Jay Lebeau, and Randy Lillard for the opportunity to work on this software development project.

References

[1] Carl B. Moyer and Roald A. Rindal. An analysis of the coupled chemically reacting boundary layer and charring ablator, part ii: Finite difference solution for the in-depth response of charring materials considering surface chemical and energy balances. Technical Report 66-7 Part II, Aerotherm, March 1967.

[2] R. M. Kendall. An analysis of the coupled chemically reactiing boundary layer and charring ablator, part v: A general approach to the thermochemical solution of mixed equilibirum-nonequilibrium , homogeneous and heterogeneous systems. Technical Report 66-7 Part V, Aerotherm, June 1968.

[3] E. P. Bartlett and R. M. Kendall. An analysis of the coupled chemically reactiing boundary layer and charring ablator, part iii: Nonsimilar solution fo the multicomponent laminar boundary layer by an integral matrix procedure. Technical Report 66-7 Part III, Aerotherm, June 1968.

[4] G.C. April, R. W. Pike, and E.G. del Valle. Modeling reaction gas flow in the char layer of an ablator. *AIAA Journal*, 9(6):1113–1119, 1971.

[5] E.G. del Valle, R. W. Pike, and G.C. April. Transport phenomena in the char zone ii: Equilibrium composition of degradation products of ablation. *Symposium on the Technology of Ablation, American Institute of Chemical Engineers*, 1967.

[6] Y.-K. Chen Frank Milos. Ablation and thermal response program for spacecraft heatshield design. *Journal of Spacecraft and Rockets*, 36(3):475–483, 1999.

[7] Mairead Stackpoole, Steve Sepka, Ioana Cozmuta, and Dean Kontinos. Post-flight evaluation of stardust sample return capsule forebody heatshield material, 2008. 46th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2008-1202.

[8] Michael Wright, Robin Beck, Karl Edquist, David Driver, Steven Sepka, Eric Slimco, William Wilcockson, Anthony DeCaro, and Helen Hwang. Sizing and margins assessment for for the mars science laboratory aeroshell thermal protection system, 2009. 41st AIAA Thermophysics Conference, AIAA Paper 2009-4231.

[9] David J. Kinney. Impact of tps recession on the orion cev aerodynamics, 2009. 47th AIAA Aerospace Sciences Meeting, AIAA Paper 2009-1102.

[10] B.F. Blackwell. Numerical prediction of one-dimensional ablation using a finite control volume procedure with exponential differencing. *Numerical Heat Transfer*, 14(1):17–34, 1988.

[11] B.F. Blackwell and R. E. Hogan. One-dimensional ablation using landau transformation and finite control volume procedure. *Journal of Thermophysics and Heat Transfer*, 8(2):282–287, 1994.

[12] R. E. Hogan, B.F. Blackwell, and R.J. Cochran. Application of moving grid control volume finite element method to ablation problems. *Journal of Thermophysics and Heat Transfer*, 10(2):312–319, 1996.

[13] Hyo-Keun Ahn, Chul Park, and Keisuke Sawada. Response of heatshield material at stagnation point of poineer-venus probe. *Journal of Thermophysics and Heat Transfer*, 16(3):432–439, 2002.

[14] Toshiyuki Suzuki, Keisuke Sawada, Tetsuya Yamada, and Yoshifumi Inatani. Thermal response of abltive test piece in arc-heated wind tunnel, 2004. 42th AIAA Aerospace Sciences Meeting and Exhibit, AIAA Paper 2004-0341.

[15] Adam J. Amar. Modeling of One-Dimensional Ablation with Porous Flow Using F inite Control Volume Procedure. Master's thesis, North Carolina State University, 2006.

[16] Adam J. Amar, Bennie F. Blackwell, and Jack R. Edwards. One-Dimensional Ablation Using a Full Newton's Method and Finite Control Volume Procedure. *AIAA Journal of Thermophysics and Heat Transfer*, 22(71-82):1, 2006.

[17] Adam J. Amar, Bennie F. Blackwell, and Jack R. Edwards. Development and Verification of a One-Dimensional Ablation Code Including Pyrolysis Gas Flow. *AIAA Journal of Thermophysics and Heat Transfer*, 23(59-71):1, 2006.

[18] J. A. Dec and R. D. Braun. Ablative thermal response analysis using the finite element method. *47th Aerospace Sciences Meeting*, January 2009.

[19] John A. Dec. *Three Dimensional Finite Element Ablative Thermal Response Analysis as Applied to Heatshield Penetration Design*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, March 2010.

[20] P. J. Roache. *Verification and Validation in Computational Science and Engineering*. Hermosa Publishers, 1998.

[21] Youcef Saad and Martin H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, 1986.

[22] P. Knupp and K. Salari. *Verification of Computer Codes in Computational Science and Engineering*. Chapman & Hall/CRC, 2003.

[23] David G. Goodwin. Cantera c++ user's guide. Technical report, California Institute of Technology, October 2002.

[24] Sanford Gordon and Bonnie J. McBride. Computer program for calculation of complex chemical equilibrium compositions and applications i. analysis. Technical Report RP-1311, NASA, October 1994.

[25] Bonnie J. McBride and Sanford Gordon. Computer program for calculation of complex chemical equilibrium compositions and applications ii. user's manual and program description. Technical Report RP-1311-P2, NASA, June 1996.

[26] Henry Darcy. Les fontaines publiques de la ville dijon. Dalmont, Paris, France, 1856.

[27] Eric B. Becker, Graham F. Carey, and J. Tinsley Oden. *Finite Elements – An Introduction*, volume 1. Prentice Hall, 1981.

[28] Benjamin S. Kirk. *Adaptive Finite Element Simulation of Flow and Transport Applications on Parallel Computers*. PhD thesis, The University of Texas at Austin, May 2007.

[29] Benjamin S. Kirk and Graham F. Carey. Development and Validation of a SUPG Finite Element Scheme fo r the Compressible Navier-Stokes Equations using a Modified Inviscid Flux Discre tization. *International Journal for Numerical Methods in Fluids*, 57(3):265 – 293, 2008.

[30] Benjamin S. Kirk, John W. Peterson, Roy H. Stogner, and Graham F. Carey. `libMesh:` A C++ Library for Parallel Adaptive Mesh Refinement/Coarsening Simulations. *Engineering with Computers*, 22(3):237–254, 2006.

[31] G. Karypis and V. Kumar. METIS unstructured graph partitioning and sparse matrix order. Technical report, University of Minnesota, Department of Computer Science, August 1995.

[32] Satish Balay, Kris Buschelman, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Barry F. Smith, and Hong Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 2.3.0, Argonne National Laboratory, April 2004.

[33] M. Utku and G. F. Carey. Boundary penalty techniques. *Computer Methods in Applied Mechanics and Engineering*, 30:103–118, 1982.

[34] V. S. Arpaci. *Conduction Heat Transfer*. Addison-Wesley, 1966.

[35] J. V. Beck, B. F. Blackwell, and C. R. St. Clair. *Inverse Heat Conduction Problems*. Wiley-Interscience, New York, NY, 1985.

[36] Adrian Bejan. *Heat Transfer*. John Wiley & Sons, 1993.

[37] Maplesoft. *Maple User Manual*, 2010.

[38] Stephen Wolfram. *The Mathematica Book*. Wolfram Media, 5th edition, 2003.

Development and Verification of the Charring Ablating Thermal Protection Implicit System Solver

Adam Amar

Ben Kirk

Nathan Calvert

AIAA Thermophysics Conference, January 2011

Outline

Software Description

Modeling Details

- Governing equations
- Linearization with complex-step method

Verification Studies

- 2D nonlinear energy equation
- 3D nonlinear energy equation MMS problem
- 3D nonlinear gas flow equation MMS problem

Complex-Step Study

Conclusions and Future Work

Software Description

Solves two and three dimensional charring ablation, heat transfer, and porous flow problems with moving mesh capability in both serial and parallel

Uses libMesh finite element library

- mesh data
- domain decomposition (METIS)
- parallel communication (MPI)
- linear system solver (PETSc)

Galerkin finite element spatial discretization on unstructured hybrid meshes

- First order Lagrangian basis functions (second order discretization)
- Gaussian quadrature

First and second order implicit time integrators

Fully implicit linear system

- Newton’s method
- Generalized minimal residual method

Adaptive mesh refinement with all element types

- 2D: quadrilateral and triangle
- 3D: tetrahedron, hexahedron, prism and pyramid

Uses Cantera for gas thermochemical and transport properties

Governing Equations

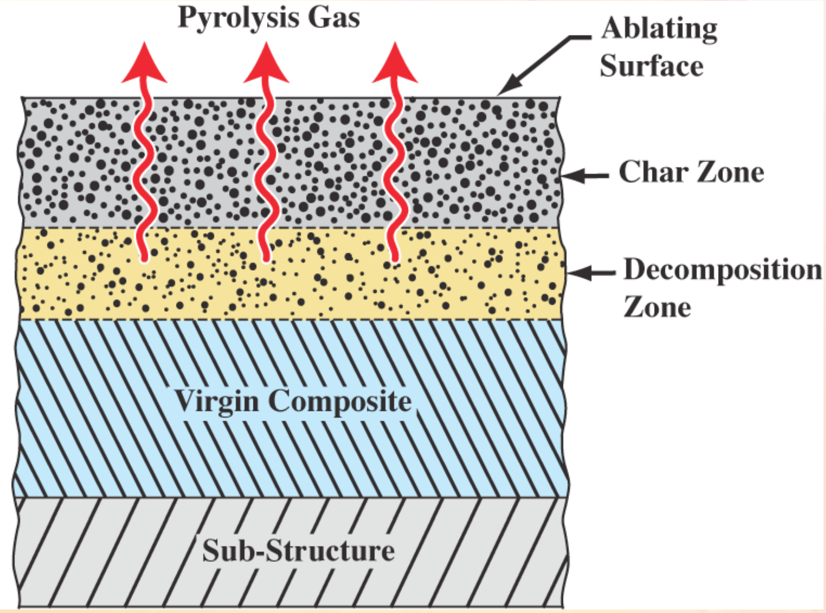
Implemented in code

$$\text{Energy}: \frac{\partial(\rho e_o)}{\partial t} - \underbrace{\nabla\cdot\tilde{\mathbf{k}}\nabla T}_{\text{Conduction}} + \underbrace{\nabla\cdot\left(\phi\rho_g h_{o_g}\mathbf{v}_g\right)}_{\text{Gas Flow}} - \underbrace{\nabla\cdot\left(\rho h_o \mathbf{v}_m\right)}_{\text{Mesh Motion}} = \underbrace{\dot{Q}}_{\text{Source}}$$

$$\text{Solid}: \frac{\partial \rho_s}{\partial t} - \underbrace{\nabla\cdot\left(\rho_s \mathbf{v}_m\right)}_{\text{Mesh Motion}} = \underbrace{\dot{\omega}_s}_{\text{Source}}$$

$$\text{Gas}: \frac{\partial(\phi\rho_g)}{\partial t} + \underbrace{\nabla\cdot\left(\phi\rho_g \mathbf{v}_g\right)}_{\text{Gas Flow}} - \underbrace{\nabla\cdot\left(\phi\rho_g \mathbf{v}_m\right)}_{\text{Mesh Motion}} = \underbrace{\dot{\omega}_g}_{\text{Source}}$$

$$\text{Darcy's Law}: \mathbf{v}_g = -\frac{\tilde{\kappa}}{\phi\mu}\nabla P$$



Verified in this study

$$\text{Energy}: \frac{\partial(\rho e_o)}{\partial t} - \underbrace{\nabla\cdot\tilde{\mathbf{k}}\nabla T}_{\text{Conduction}} + \underbrace{\nabla\cdot\left(\phi\rho_g h_{o_g}\mathbf{v}_g\right)}_{\text{Gas Flow}} - \underbrace{\nabla\cdot\left(\rho h_o \mathbf{v}_m\right)}_{\text{Mesh Motion}} = \underbrace{\dot{Q}}_{\text{Source}}$$

$$\text{Solid}: \frac{\partial \rho_s}{\partial t} - \underbrace{\nabla\cdot\left(\rho_s \mathbf{v}_m\right)}_{\text{Mesh Motion}} = \underbrace{\dot{\omega}_s}_{\text{Source}}$$

$$\text{Gas}: \frac{\partial(\phi\rho_g)}{\partial t} + \underbrace{\nabla\cdot\left(\phi\rho_g \mathbf{v}_g\right)}_{\text{Gas Flow}} - \underbrace{\nabla\cdot\left(\phi\rho_g \mathbf{v}_m\right)}_{\text{Mesh Motion}} = \underbrace{\dot{\omega}_g}_{\text{Source}}$$

$$\text{Darcy's Law}: \mathbf{v}_g = -\frac{\tilde{\kappa}}{\phi\mu}\nabla P$$

Linearization

Linearization in iteration space

$$R_i^{\nu+1} = R_i^{\nu} + \sum_{j=1}^{N} \left\{ \left. \frac{\partial R_i}{\partial T_j} \right|^{\nu} \Delta T_j + \left. \frac{\partial R_i}{\partial \left(\phi \rho_g\right)_j} \right|^{\nu} \Delta \left(\phi \rho_g\right)_j \right\} + \text{higher order terms}$$

Jacobian Terms

Analytically take the derivatives of the residual and implement resulting expressions into code

- Typically approximated for very complicated residual equations
- Prone to errors in both derivation and implementation
- Approximation and errors degrade convergence rate of nonlinear system solver

Finite-difference approximations are also common

- Eliminates derivation errors
- Approximations degrade convergence rates of nonlinear system solver
- Choice of step-size is not straightforward
- Second order central difference requires 2 perturbed residual evaluations

Complex-Step Method

Taylor series with complex-step

$$R\left[T_j+i\Delta T_j,\left(\phi\rho_g\right)_j\right]=R\left[T_j,\left(\phi\rho_g\right)_j\right]+i\frac{\partial R}{\partial T_j}\Delta T_j-\frac{1}{2}\frac{\partial^2 R}{\partial T_j{}^2}\left(\Delta T_j\right)^2-\frac{i}{6}\frac{\partial^3 R}{\partial T_j{}^3}\left(\Delta T_j\right)^3$$

Real part gives residual

$$R\left[T_j,\left(\phi\rho_g\right)_j\right]=\Re\left\{R\left[T_j+i\Delta T_j,\left(\phi\rho_g\right)_j\right]\right\}+O\left[\left(\Delta T_j\right)^2\right]$$

Imaginary part gives Jacobian

$$\frac{\partial R}{\partial T_j}=\frac{\Im\left\{R\left[T_j+i\Delta T_j,\left(\phi\rho_g\right)_j\right]\right\}}{\Delta T_j}+O\left[\left(\Delta T_j\right)^2\right]$$

Truncation error with extremely small perturbations results in exact residual and Jacobian

Advantages:

- No Jacobian derivation necessary
- Easy addition of new models
- Reduces possibility of bugs
- Reduces development time
- Optimizes convergence
- Choice of perturbation size is simple
- Get residual and Jacobian from one evaluation

Disadvantages:

- Extra time spent doing complex arithmetic
- Incompatible with external libraries

Verification and Validation

VERIFICATION: Are we solving the equations correctly?

- Do the spatial and temporal discretizations exhibit the theoretical orders of accuracy?
- Does the Newton solver converge at the theoretical rate?
- ***Strictly a mathematical exercise***
- Does not need to bear any semblance to the reality of the problems the code intends to solve
- Done through comparison of simulation solutions with known exact solutions
  - Analytic solutions to PDE(s)
  - Method of Manufactured Solutions
- Should be done for all boundary conditions, governing equation terms, and code options
- Should precede validation
- Neither extensive code use nor code-to-code comparisons constitute verification

VALIDATION: Are we solving the correct equations?

- Do the model equations represent physical reality?
- Done through comparison of simulation solution with test data
- May have to validate and develop best practices for several different regimes/problem classes

Verification Strategy

Typically spatial and temporal orders of accuracy are independently verified

$$DE = a\Delta t^{p} + b\Delta h^{q} \Rightarrow \frac{\partial[\log(DE)]}{\partial[\log(\Delta t)]} = p = 2$$

Simultaneous refinement can be used to verify both temporal and spatial accuracy

$$DE = \underbrace{\left(1 + \frac{a\Delta t^{p}}{b\Delta h^{q}}\right)}_{\text{Hold Constant}} b\Delta h^{q} \Rightarrow \boxed{\frac{\partial[\log(DE)]}{\partial[\log(\Delta h)]} = q = 2}$$

Newton’s method should exhibit quadratic convergence

$$E^{\nu+1} = a\left(E^{\nu}\right)^{2} \Rightarrow \boxed{\frac{\partial\left[\log\left(E^{\nu+1}\right)\right]}{\partial\left[\log\left(E^{\nu}\right)\right]} = 2}$$

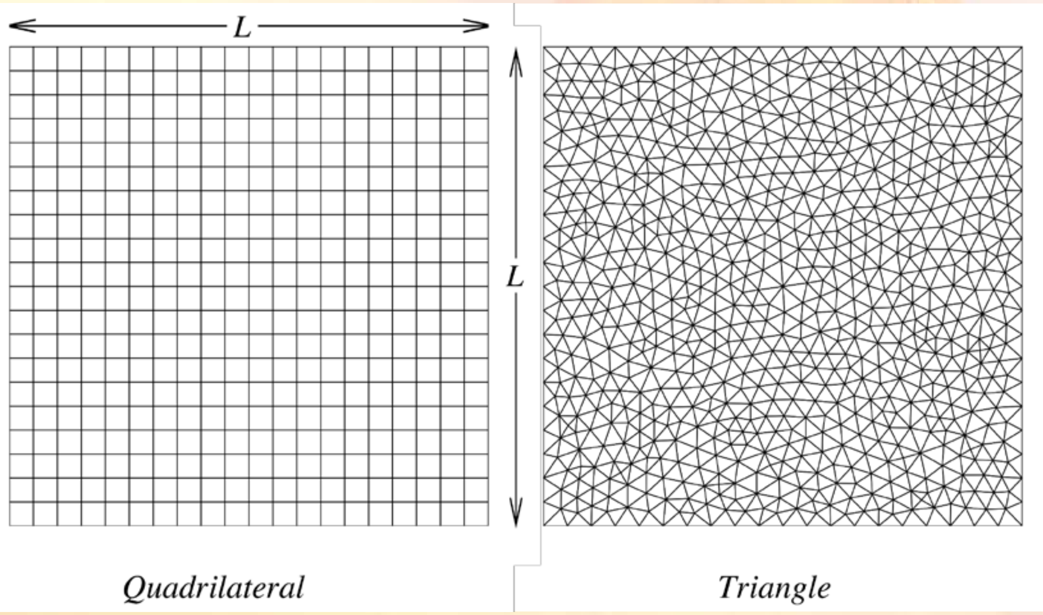
2-D Energy Equation: Analytic Solution (I)

Uniform density specimen exposed to constant heat flux on one face

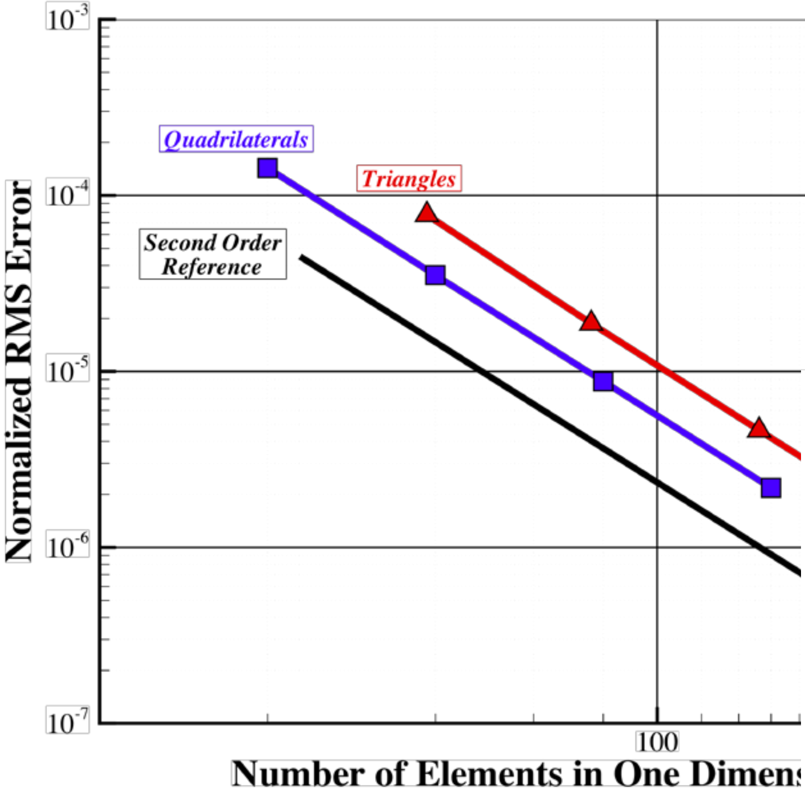
- Adiabatic on three other faces
- Thermal properties linear in temperature
- Solution is 1D in nature, but suitable for 2D and 3D verification with unstructured solver

Grid refinement study

- Quadrilateral and triangle meshes
- 4 grid levels with space/time refinement
- Coarse grids show here

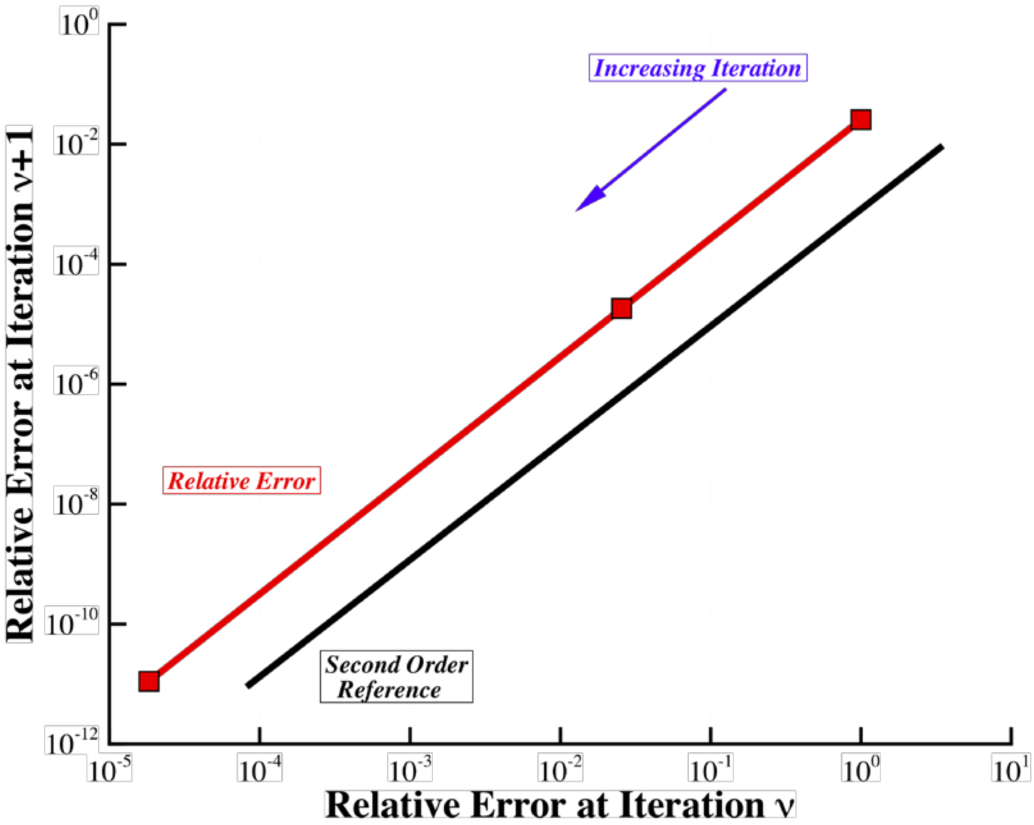


2-D Energy Equation: Analytic Solution (II)



Grid Convergence

Nonlinear Convergence



Method of Manufactured Solutions (I)

Design problem known exact solution

- exercise multiple terms
- exercise multiple boundary conditions
- arbitrary solution domain
- one solution can be used for multiple verification problems
- typically non-physical

Specify PDE(s) to be verified

$$\rho C_p \frac{\partial T}{\partial t} - \boldsymbol{\nabla}\cdot(k\boldsymbol{\nabla}T) = \dot{Q}$$

Prescribe a solution

$$T(\boldsymbol{x},t) = \theta\cos(A_x x + A_t t)\cos(B_y y + B_t t)\cos(C_z z + C_t t)\cos(D_t t) + T_o$$

Select property models

$$k\left(T\right) = k_0 + k_1\left(\frac{T-T_o}{\theta}\right) + k_2\left(\frac{T-T_o}{\theta}\right)^2$$

$$C_p\left(T\right) = C_{p_0} + C_{p_1}\left(\frac{T-T_o}{\theta}\right) + C_{p_2}\left(\frac{T-T_o}{\theta}\right)^2$$

Method of Manufactured Solutions (II)

Substitute solution and property models into governing equations and solve for source term

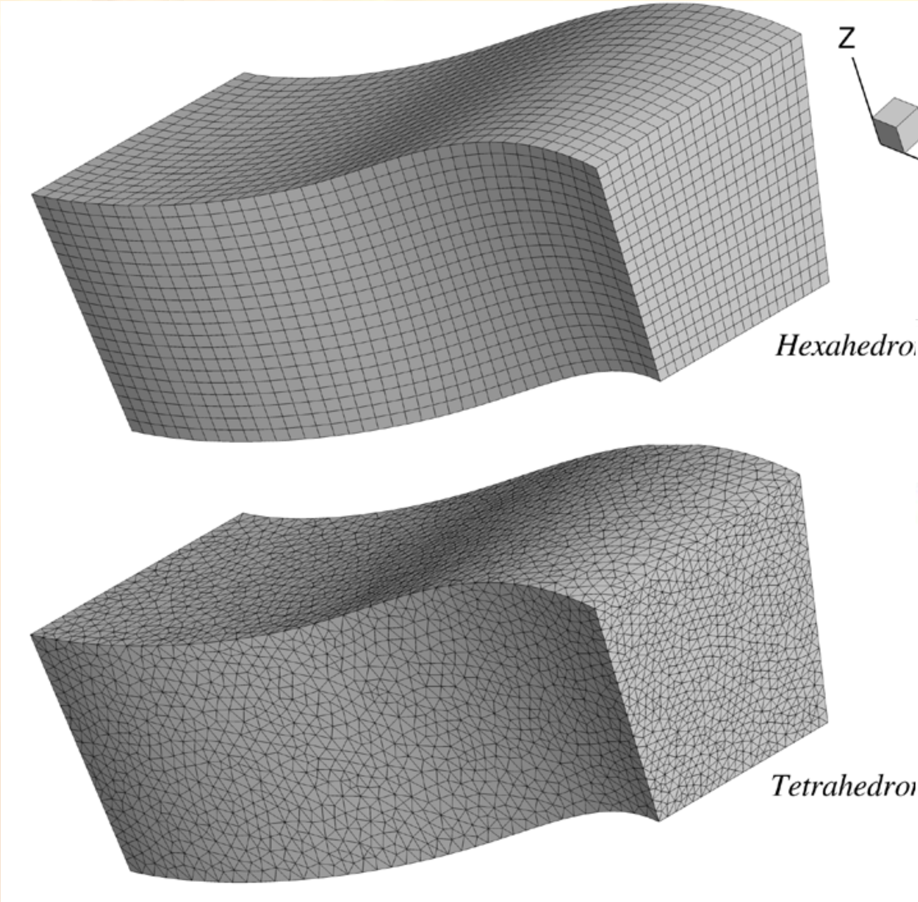
$$
\begin{aligned}
\dot{Q}\,(x,t) =\rho\,[\, & C_{p_0} + C_{p_1} \cos(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& + C_{p_2} \cos^2(A_x x + A_t t) \cos^2(B_y y + B_t t) \cos^2(C_z z + C_t t) \cos^2(D_t t)\ ] \\
& \{ -\theta\, [\ A_t \sin(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& + B_t \cos(A_x x + A_t t) \sin(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& + C_t \cos(A_x x + A_t t) \cos(B_y y + B_t t) \sin(C_z z + C_t t) \cos(D_t t) \\
& + D_t \cos(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \sin(D_t t)\ ]\ \} \\
& - \{\ [\ k_0 + k_1 \cos(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& + k_2 \cos^2(A_x x + A_t t) \cos^2(B_y y + B_t t) \cos^2(C_z z + C_t t) \cos^2(D_t t)\ ] \\
& \left[-\theta \left(A_x^2 + B_y^2 + C_z^2\right) \cos(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t)\right] \\
& - \theta\, [\ A_x \sin(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& (\ - k_1 A_x \sin(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& + 2k_2 A_x \sin(A_x x + A_t t) \cos(A_x x + A_t t) \cos^2(B_y y + B_t t) \cos^2(C_z z + C_t t) \cos^2(D_t t)\ ) \\
& + B_y \cos(A_x x + A_t t) \sin(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& (\ - k_1 B_y \cos(A_x x + A_t t) \sin(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) \\
& + 2k_2 B_y \cos^2(A_x x + A_t t) \sin(B_y y + B_t t) \cos(B_y y + B_t t) \cos^2(C_z z + C_t t) \cos^2(D_t t)\ ) \\
& + C_z \cos(A_x x + A_t t) \cos(B_y y + B_t t) \sin(C_z z + C_t t) \cos(D_t t) \\
& (\ - k_1 C_z \cos(A_x x + A_t t) \cos(B_y y + B_t t) \sin(C_z z + C_t t) \cos(D_t t) \\
& + 2k_2 C_z \cos^2(A_x x + A_t t) \cos^2(B_y y + B_t t) \sin(C_z z + C_t t) \cos(C_z z + C_t t) \cos^2(D_t t)\ )\ \}
\end{aligned}
$$

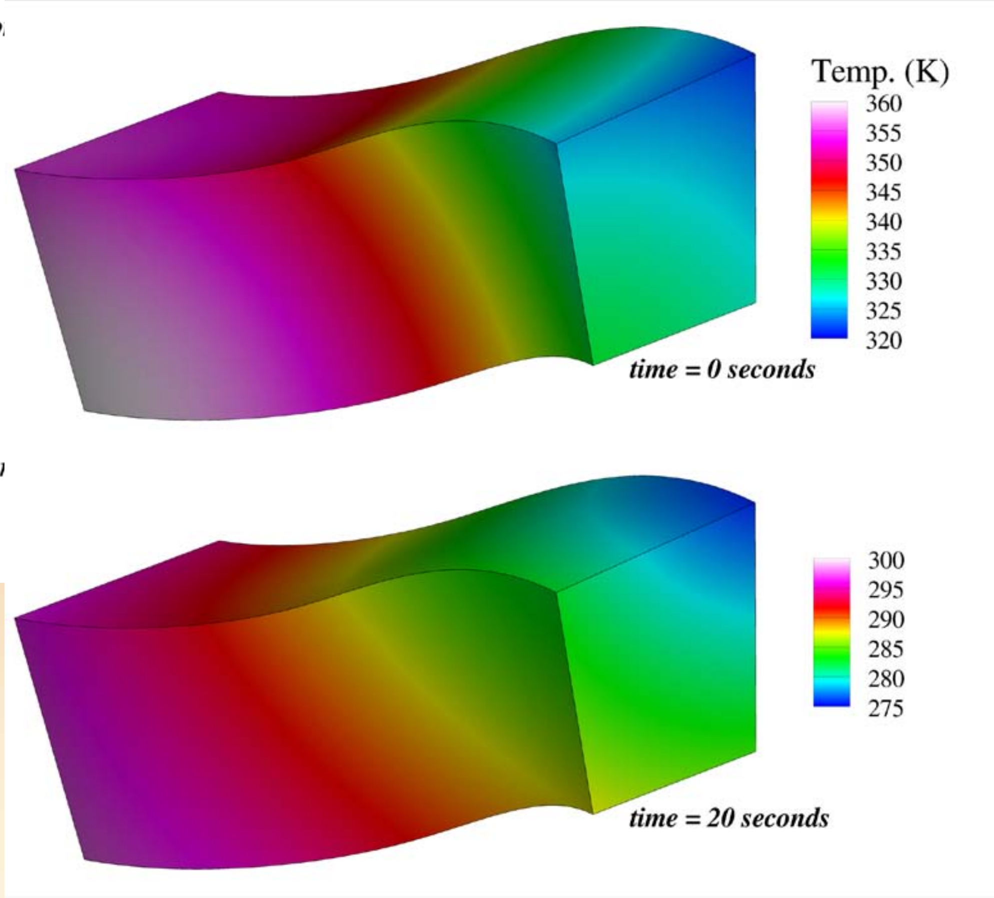
Implement source term in source code

Perform verification studies

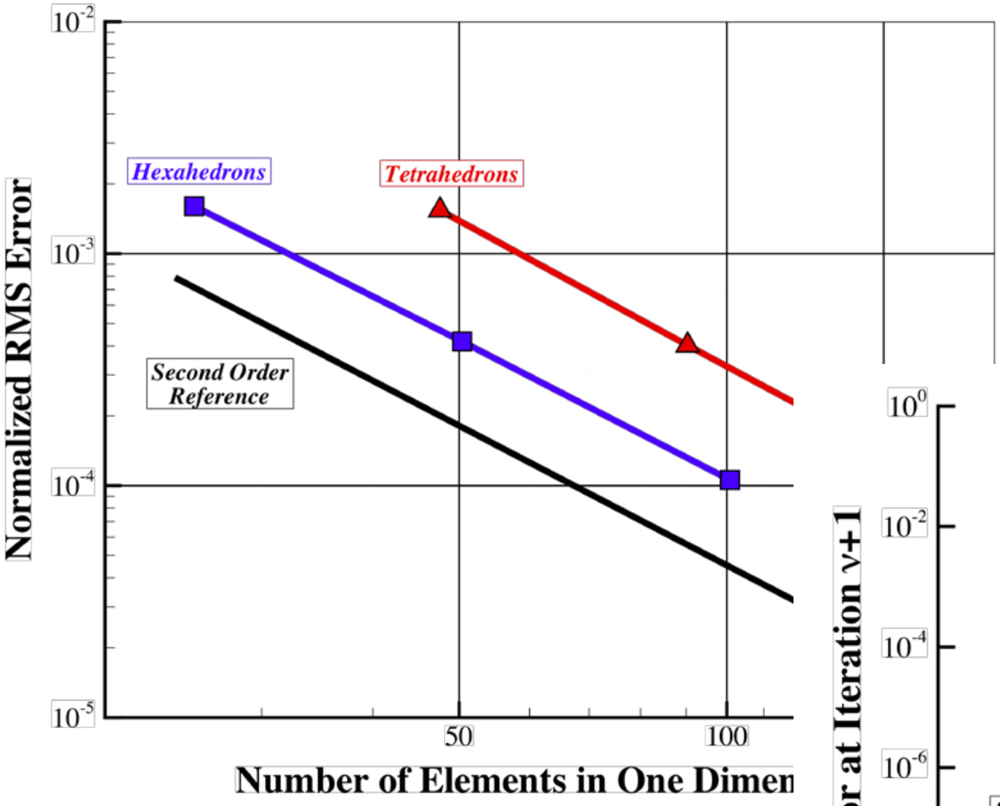
- 1D, 2D, or 3D
- Transient or steady state
- Linear or nonlinear

3-D Energy Equation: MMS (I)



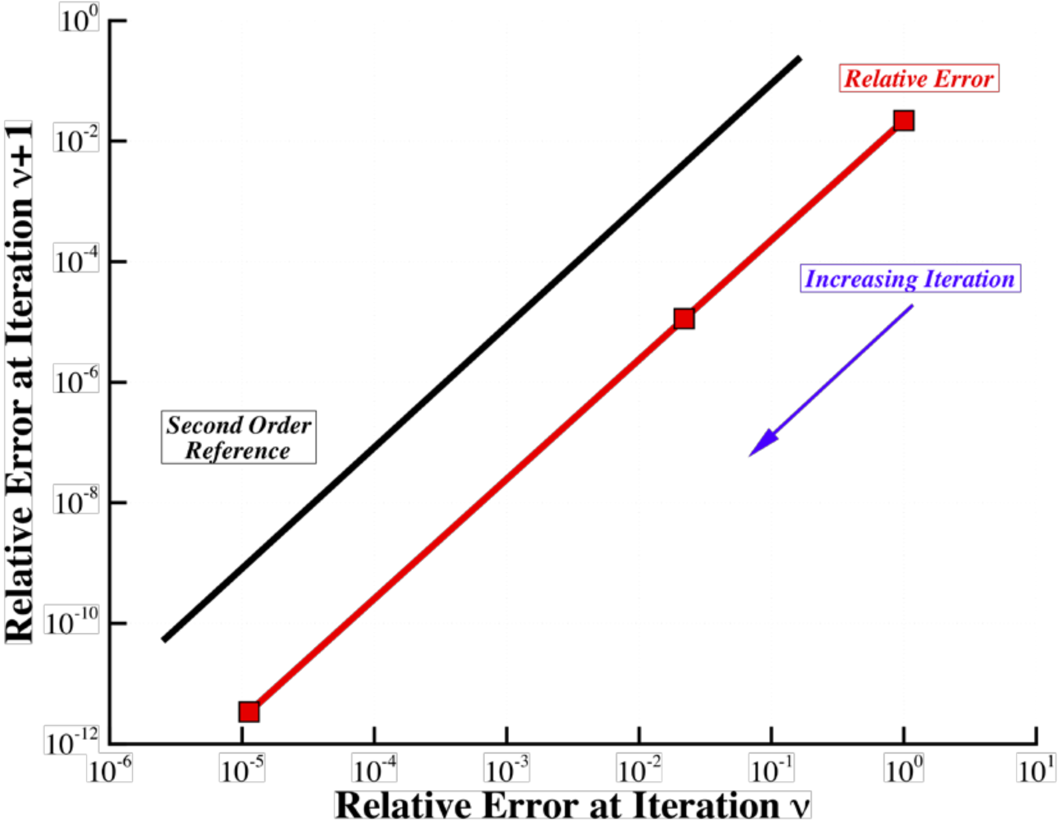


3-D Energy Equation: MMS (II)



Grid Convergence

Nonlinear Convergence



3-D Gas Flow Equation: MMS (I)

Gas Flow Equation with constant porosity, permeability, and viscosity

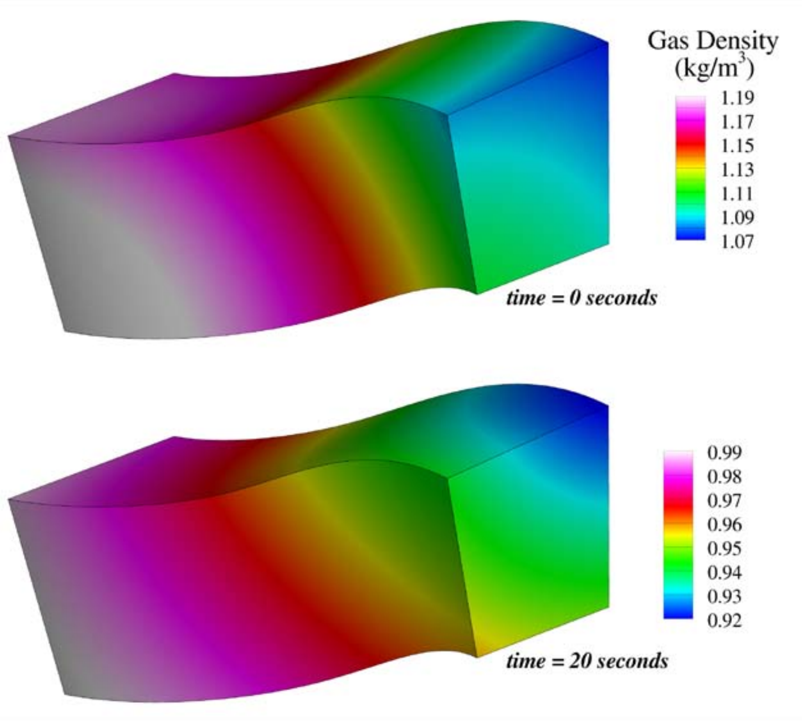
$$\phi \frac{\partial \rho_g}{\partial t} - \frac{\kappa R T}{\mu} \boldsymbol{\nabla} \cdot \left(\rho_g \boldsymbol{\nabla} \rho_g \right) = \dot{\omega}_g$$

Manufactured solution

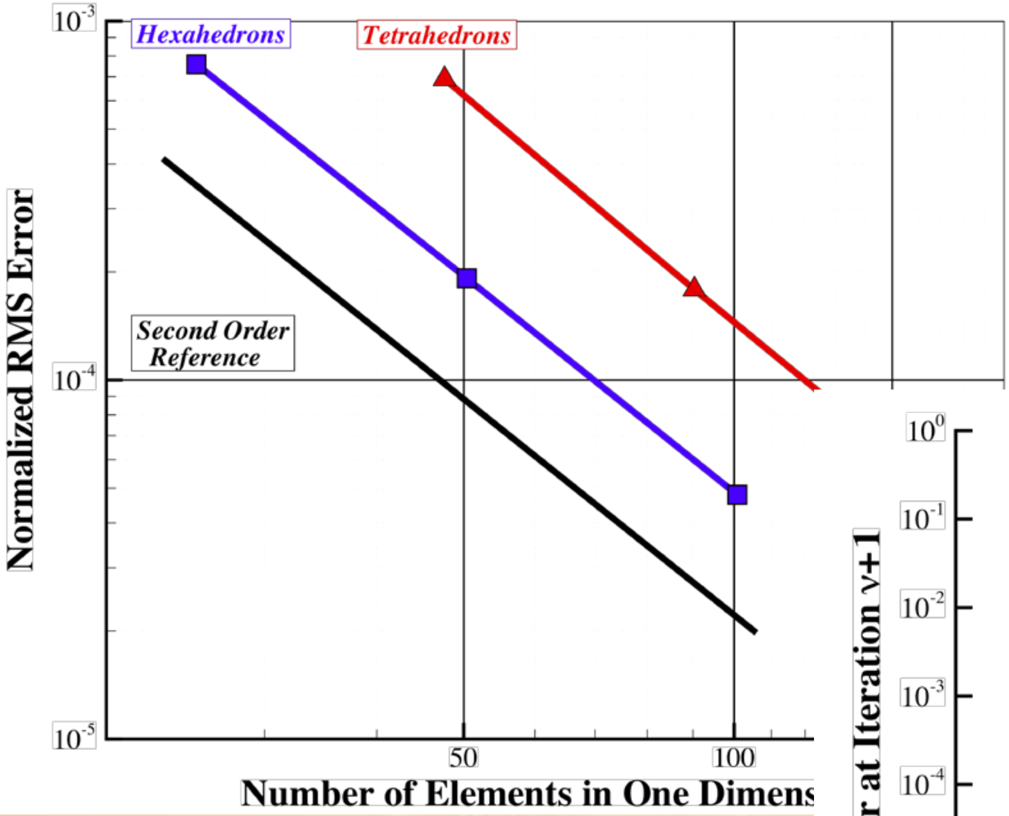
$$\rho_g(\boldsymbol{x}, t) = \theta \cos(A_x x + A_t t) \cos(B_y y + B_t t) \cos(C_z z + C_t t) \cos(D_t t) + \rho_{g_o}$$

Resulting source term

$$\begin{aligned}
\dot{\omega}_g \left(\boldsymbol{x}, t\right) = -\phi\theta \, [& A_t \sin\left(A_x x + A_t t\right) \cos\left(B_y y + B_t t\right) \cos\left(C_z z + C_t t\right) \cos\left(D_t t\right) \\
& + B_t \cos\left(A_x x + A_t t\right) \sin\left(B_y y + B_t t\right) \cos\left(C_z z + C_t t\right) \cos\left(D_t t\right) \\
& + C_t \cos\left(A_x x + A_t t\right) \cos\left(B_y y + B_t t\right) \sin\left(C_z z + C_t t\right) \cos\left(D_t t\right) \\
& + D_t \cos\left(A_x x + A_t t\right) \cos\left(B_y y + B_t t\right) \cos\left(C_z z + C_t t\right) \sin\left(D_t t\right) \,] \\
- \frac{\kappa R T}{\mu} \{ & -\rho_{g_o} \theta \left(A_x^2 + B_y^2 + C_z^2\ \right) \cos\left(A_x x + A_t t\right) \cos\left(B_y y + B_t t\right) \cos\left(C_z z + C_t t\right) \cos\left(D_t t\right) \\
& -\theta^2 \left(A_x^2 + B_y^2 + C_z^2\ \right) \cos^2\left(A_x x + A_t t\right) \cos^2\left(B_y y + B_t t\right) \cos^2\left(C_z z + C_t t\right) \cos^2\left(D_t t\right) \\
+\theta^2 \cos^2\left(D_t t\right) [& A_x^2 \sin^2\left(A_x x + A_t t\right) \cos^2\left(B_y y + B_t t\right) \cos^2\left(C_z z + C_t t\right) \\
& + B_y^2 \cos^2\left(A_x x + A_t t\right) \sin^2\left(B_y y + B_t t\right) \cos^2\left(C_z z + C_t t\right) \\
& + C_z^2 \cos^2\left(A_x x + A_t t\right) \cos^2\left(B_y y + B_t t\right) \sin^2\left(C_z z + C_t t\right) \,] \, \}
\end{aligned}$$

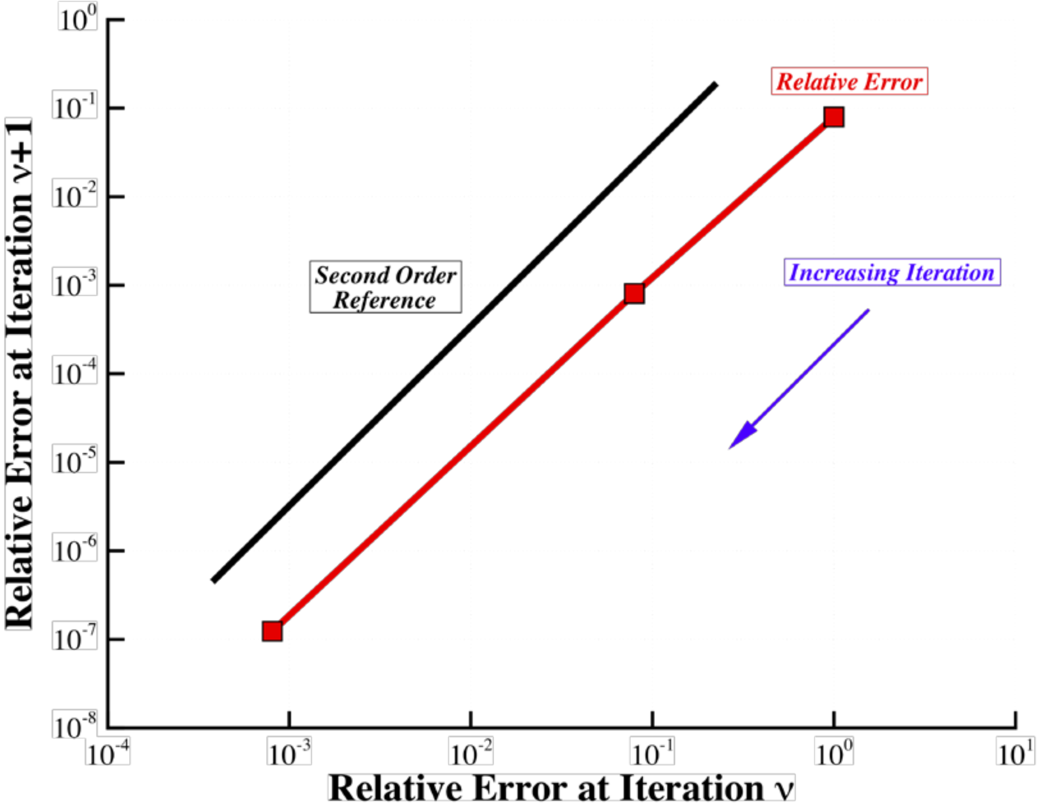


3-D Gas Flow Equation: MMS (II)



Grid Convergence

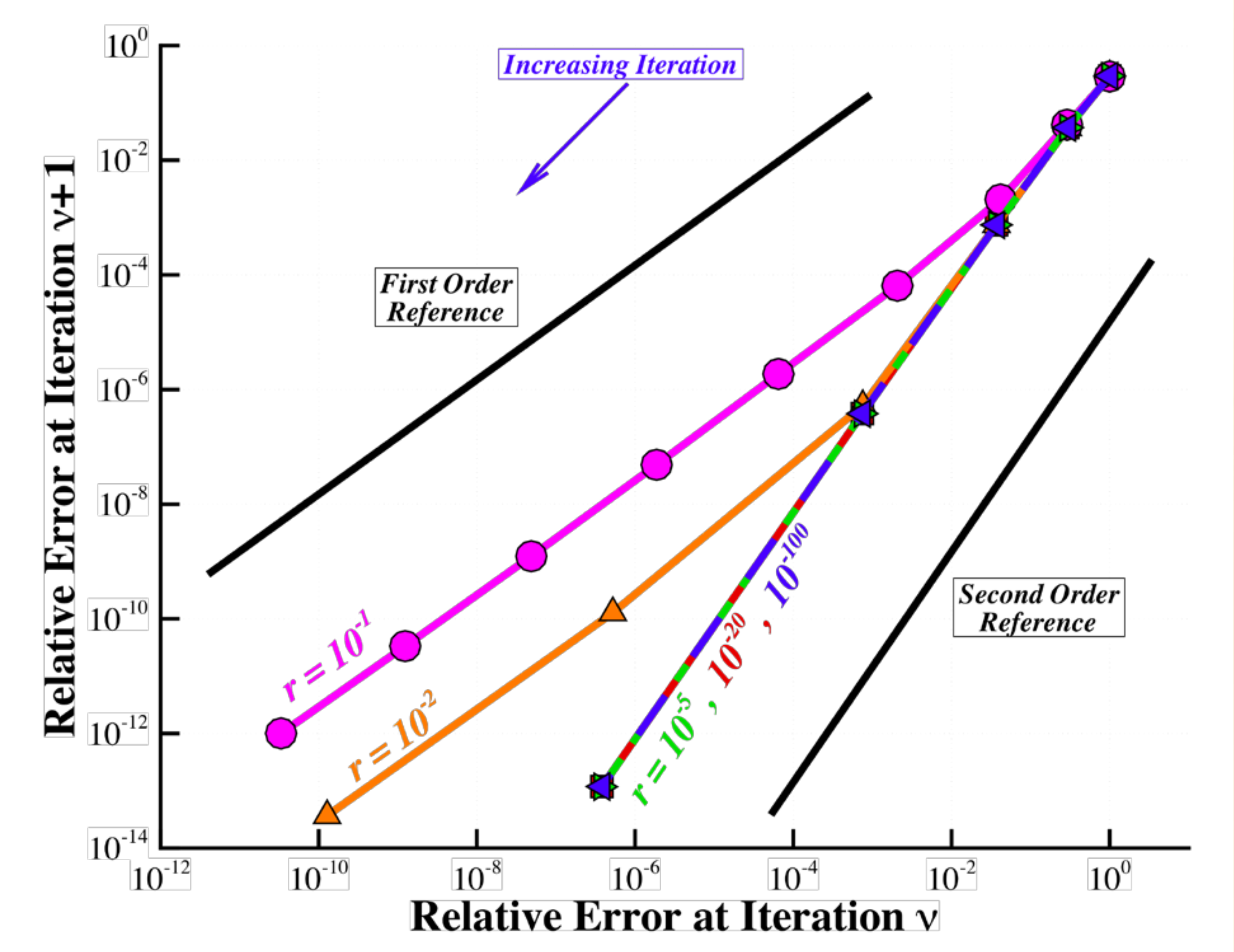
Nonlinear Convergence



Complex-Step Size Study

What is the best choice for the step size? $\Delta T = rT$

Solved 1 time step of nonlinear conduction problem with 5 choices of perturbation factor



Large choices still converge but at degraded rates

Very small choices present no numerical issues

Broad range of acceptable choices to give optimal convergence and reduce computational time

Choice of perturbation factor simple compared to finite difference

Conclusions and Future Work

Most stationary mesh terms in the software have been verified to exhibit the theoretical orders of accuracy

- Verified both 2D and 3D element types
- Used analytic solutions and MMS

Implemented to complex-step method to calculate exact Jacobians

- Verified convergence rates of nonlinear solver
- Examined sensitivity to perturbation factor

Future verification

- Coupled equation MMS problem
- Moving mesh terms
- Adaptive mesh refinement cases

Modeling additions and software improvements

- Inverse capability
- Sensitivity analysis and uncertainty propagation
- Higher fidelity physical models (thermal and chemical non-equilibrium, etc.)
- Improved mesh motion algorithms
- Profiling and optimization

Acknowledgements

- Ben Blackwell
- Kemelli Estacio-Hiroms
- Brandon Oliver
- Chuck Campbell
- Amy Cassady
- Jay Lebeau
- Randy Lillard