ANS MC2015 - Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method · Nashville, Tennessee · April 19–23, 2015, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2015)

INITIAL 1-D SINGLE PHASE LIQUID TRANSIENT VERIFICATION OF COBRA-TF

Chris Dances and Dr. Maria Avramova
Department of Mechanical and Nuclear Engineering
The Pennsylvania State University
137 Reber Building, University Park, PA, 16802, USA
cad39@psu.edu; mna109@psu.edu

Dr. Vince Mousseau
Computer Science Research Institute
Sandia National Laboratories
1450 Innovation Parkway, Albuquerque, NM 87123, USA
vamouss@sandia.gov

ABSTRACT

Abstract . . . When in the course of human events . . .

Key Words: List no more than five key words

1 INTRODUCTION

For the past several decades, the primary focus in nuclear engineering within the United States has been focused on light water reactors (LWR). Commercially, all nuclear reactors are either boiling water reactors (BWR) or pressurized water reactors (PWR). Correct computation of the thermal hydraulics within the reactor core leads to efficient design and accuracy in the safety analysis. A popular subchannel code for modelling the hydrodynamics with in the reactor core is COBRA-TF. This FORTRAN based code solves 8 conservation equations for liquid, entrained droplet, and vapor phases in 3-D dimmensions [1]. A 1-D residual formulation of the code has been created. This paper outlines an intitial verification of the original verion of code as well as the residual version of the code. The verification problem is a single phase 1-D channel with transient inlet density and mass flow rate. The problem will undergo a Richardson's extrapolation in the temporal and spatial domains to verify the convergence and order of accuracy of the error. The study of the order of accuracy is considered one of the more rigourous verification criteria [2].

2 COBRA-TF

The thermal hydraulics of a LWR core is an important part of nuclear reactor desigin. COBRA-TF solves 8 conservation equations for liquid, entrained droplet, and vapor phases of water boiling within the rod structure of a LWR reactor core [1]. Currently, the conservation equations analytically reduce into a pressure matrix in a semi-implicit method with rod temperatures solved for explicitly. The residual formulation of the code currently solves the 1-D single phase liquid conservation equations and calculated variables in a residual formulation. While it has the ability to solve the code semi-implicitly or implicitly, only the semi-implicit solution method is considered in this paper. This residual formulation should allow for easier and more in depth verification analysis. This paper details the intitial comparison of the residual formulation to the original code.

2.1 Software Quality Assurance

Software quality assurance is a set of tools and procedures that helps ensure that the software is reliable. COBRA-TF is managed by Github repository setup and maintained by CASL. An exensive test matrix is run before each major push to ensure that the code meets the specificed requirements. The test matrix consists of unit tests, code coverage runs, validation problems, and validation problems. The code documentation consists of a theory manual, a users manual, a developers manual, and a validation manual. Further work might involve using autodocumentation tools to keep an up to date developers manual. This paper will be the begining of a verification manual, integrating this verificaiton problem directly into the test matrix.

2.2 1-D Single Phase Liquid Conservation Equations

The finite volume structure in COBRA-TF in figure 1 is for a one-dimmensional channel in the axial direction with n number of cells. The first and last cells at 0 and n + 1 are ghost cells and act as the boundary conditions for the problem. Pressure, enthalpy, and density are averaged over the

cell volume and are located at the center of the cell. Mass flow rate and velocity are located at the faces in between cells. The cells are represented with an index i, and the faces with indexes of $i + \frac{1}{2}$ or $i - \frac{1}{2}$. This project will initially focus on this 1-D configuration. Usually the code is 3-D, with channels connecting to each other in two more dimmensions.

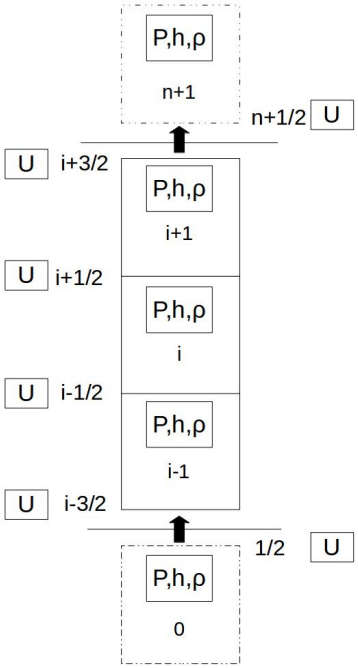

Figure 1. The finite volume structure for COBRA-TF

The single phase Euler partial differential equations for mass (1), momentum (2), and energy (3) corresond to the unknown variables density ρ, velocity u, pressure P, and enthalpy h. The first terms in each of the equations are temporal terms. The rest of the terms are steady state spatial terms.

$$\frac{\partial \rho}{\partial t} + \bigtriangledown \rho u = 0 \tag{1}$$

$$\frac{\partial \rho u}{\partial t} + \bigtriangledown \rho u^2 + \bigtriangledown P - \rho g = 0 \tag{2}$$

$$\frac{\partial \rho h}{\partial t} - \frac{\partial P}{\partial t} + \bigtriangledown (\rho u h) = 0 \tag{3}$$

2.3 Residual Formulation and Jacobian Construction

A residual is simply the difference between the value at some future time $n + 1$ and the value at the current iteration k. This can be applied to desired variables and equations. For example, the

residual for density, $\delta\rho_i$, is difference between iterate levels $n+1$ and k, $\rho_i^{n+1} - \rho_i^k$. The residuals for the equations are determined by susbsituting the residuals into the discretized equations, which should effectively change all $n+1$ into k. Each cell will have three residual variables and three residual equations. For the entire solution, we will then have a residual variable array δX, and a residual function array $F(X)$ which defines a linear system $J\delta X = -F(X)$.

The Jacobian matrix is defined as the derivative of each response of the function F_j with respect to each variable X_i. The derivative can be calculated numerically as shown by equation (4) where ϵ is a small numerical value. For COBRA-TF the equations are linear, and this numerical approximation of the Jacobian matrix is exact. This should produce the same jacobian matrix that COBRA-TF currently generates analytically.

$$J_{i,j} = \frac{\partial F_j(X)}{\partial X_i} \approx \frac{F_j(X_i + \epsilon) - F_j(X)}{\epsilon} \tag{4}$$

To build the jacobian matrix, an object oriented class was created that contains three arrays. An array that points to the residual functions, an array that points to the position within a target variable arrray, and an array that has the index that the function is to be evaluated at. These lists can be appended to in any order, but have to be appended all at the same time so that variables and functions must correspond with each other. Then to construct the jacobian matrix, the residual function and residual variable arrays can each be looped over to numerically build the jacobian matrix as seen in figure 2.

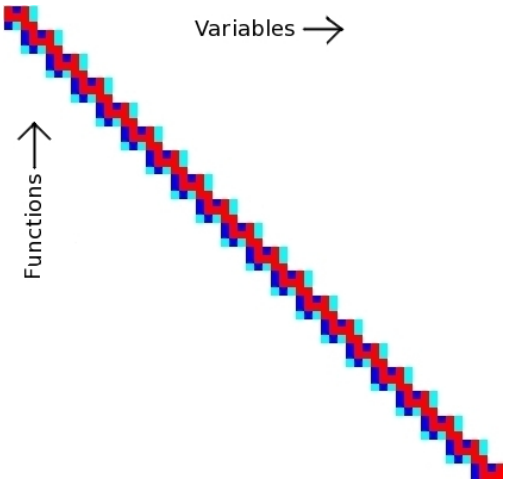

Figure 2. Strucutre of the jacobian matrix for single phase liquid

3 ISOKINETIC SINE WAVE ADVECTION

Code verification is the set of procedures set in place to ensure that the code was written properly. The procedures can use the following as code verification criteria from least to most rigorous are expert judgement, error quantification, consistencey / convergence, and order of accuruacy [3].

For this work the a Richardson Extrapolation will be used to check for convegence and order of accruacy of the error in space and time. As shown by the modified equation analysis in the previous section, the spatial and temporal order of accuracies should be 1.

3.1 Problem Setup

The verication problem is defined as a single horizontal channel problem the base parameters listed in table I. The problem will have a fixed channel area and perimeter across the entire height of the channel with no grid spacers. The velocity and pressure are assumed to be constant, but small fluctuations may occur due to coding mistakes or nuermical noise. The channel geometry and operating conditions are taken to be approximate a standard PWR. The inlet of the channel has a constant velocity with a fluctuating enthalpy that corresponds to a 37.5 °C temperature change. The length of the transient was defined to be quadrouple the time needed for the liquid at the inlet to advect to the outlet.The frequencey of the sine wave was defined so generate a full period of a spatial wave across the height of the channel.

Table I. Problem Parameters

Parameter	Symbol	Value	Unit
Axial Height	H	3.6586	m
Channel Area	A_{ch}	4.94E-005	m^2
Wetted Perimeter	P_w	1.49E-002	m
Velocity	V_o	7.35	$\frac{m}{s}$
Pressure	P_o	155.00	bar
Temperature 1	T_1	289.500	°C
Temperature 2	T_2	327.00	°C
Enthalpy 1	h_1	1281.55	$\frac{kJ}{kg}$
Enthalpy 2	h_2	1497.21	$\frac{kJ}{kg}$
Mass Flow Rate 1	$\dot{m}_1$	0.2713	$\frac{kg}{s}$
Mass Flow Rate 2	$\dot{m}_2$	0.2399	$\frac{kg}{s}$
Final Time	t_f	2.00	sec
Wave Frequency	ω	1.00	Hz

The lookup table to vary the inlet enthalpy h and inlet mass flow rate $\dot{m}$ use smooth trigonometric functions given in equations 5 and 6. The equations assume constant axial spacing Δx and time step size Δt where i and j are the spatial and temporal index respectively. These equations should also behave as the known solutions throughout the entire domain of the problem. The enthalpy and mass flow rate should vary proportionally to the density as to create an isokinetic boundary condition at the inlet. However this is dependent on the steam tables used. A python script was used to generate the data tables according to trignometric equations using lookup tables that mimic the IAPWS-97 steam tables used by the code [4].

$$h(i,j) = \frac{1}{2}\left((h_1 + h_2) + (h_1 - h_2)cos\left(\omega\left(j\Delta t + \frac{i\Delta x}{V_o}\right)\right)\right) \tag{5}$$

$$\dot{m}(i,j) = \frac{1}{2}\left((\dot{m}_1 + \dot{m}_2) + (\dot{m}_1 - \dot{m}_2)cos\left(\omega\left(j\Delta t + \frac{i\Delta x}{V_o}\right)\right)\right) \tag{6}$$

The comparison between the data table and the output in CTF are shown for enthalpy and mass flow rate in figures 3 and 4 respectively. The CTF output was read from hdf5 data files at each point in time, wich omitted the actual ghost cell where these values were applied. The CTF values are located at the nearest node to the inlet, and therefore will be slightly out of phase to the exact values but this effect is small in the figure.

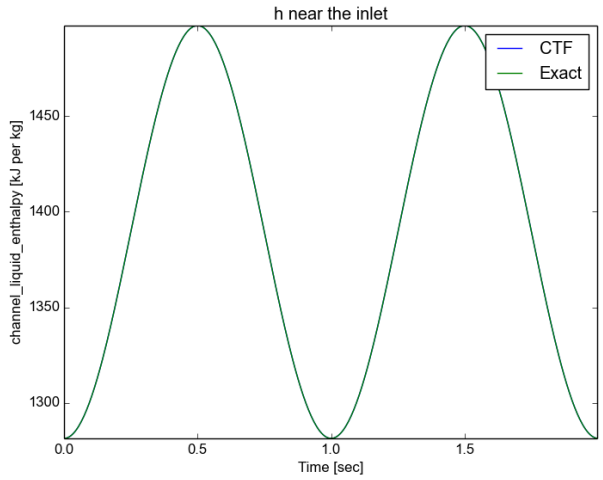

Figure 3. Enthalpy Near the Inlet and the Analytical Solution

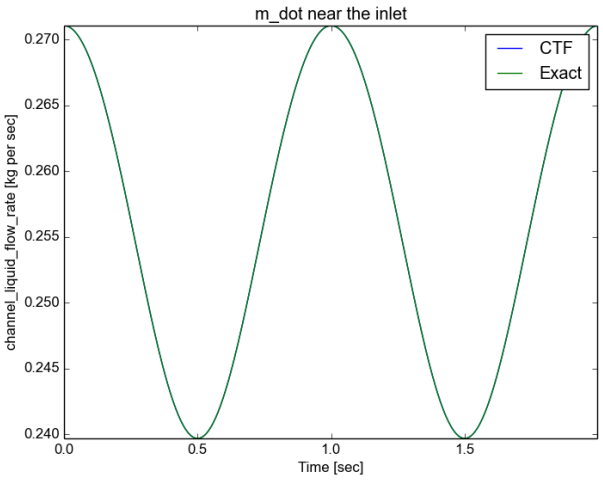

Figure 4. Density Near the Inlet and the Analytical Solution

For the original version of COBRA-TF there is a small discrepancy in the way the density is calculated at the inlet that causes the velocity to be non-consatnt. This is considered small for this problem and should not greatly affect the order of accuracy. The residual formulation was coded in

such a way as to avoid this problem and has considerably less fluctation in the inlet velocity. The hdf5 output files allow for a high level of precision of the code, reducing round off error in the output.

3.2 Code Convergence

The current version of COBRA-TF uses global code convergance criteria that are used to estimate the eror assocaited with the solution of the code for transient and steady state problems as shown in figure 5. For this problem, the solid energy storate is zero since there are not any heat structures present. The fluctuating values represent differences between the enrgy and mass entering and leaving the system.

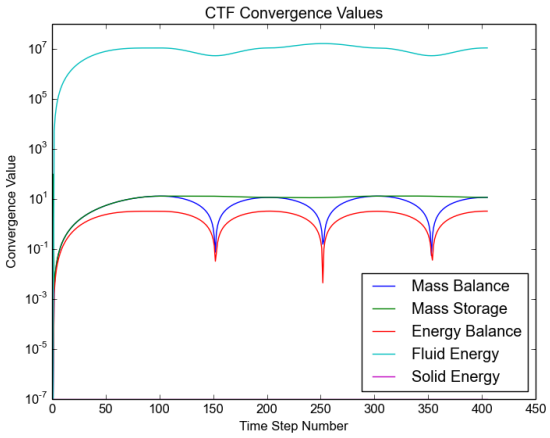

Figure 5. Code Convergence Criteria for the original version of COBRA-TF

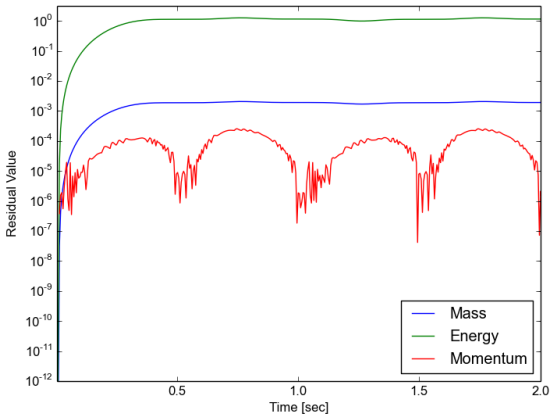

Figure 6. Sumnation of the residuals for the residual version of COBRA-TF

The residual formulation prints out the sumnation of the residuals functions across the domain to an output file for eac time step and can be seen in figure 6. The mass balance and energy balances present in the code convergeance criteria are much smaller for the residual formulation. These

residuals provide a much better indication as to the level of error present in the solution of the system. They can even be expanded to provide convergence information at every position, variable, and equation.

3.3 Modified Equation Analysis

The original mass balance equation can be re-written to look like equation 7. Using upwinding, the finite difference can be written to look like equation 8. A second order Taylor series approximation can be used for ρ_i^{n+1} and ρ_{i-1}^n as shown in equations 9 and 10 respectively. The higher order terms ($O(\Delta x^2, \Delta t^2)$) are not taken into account for this approximation. The Taylor series approximations can then be substituted into 8 to yield 11. This is the beginning of the modified equation analysis. The goal will be to isolate the original PDE and define the truncation error.

$$\frac{\partial \rho}{\partial t} + U_0 \frac{\partial \rho}{\partial x} = 0 \tag{7}$$

$$\frac{\rho_i^{n+1} - \rho_i^n}{\Delta t} + U_0 \frac{\rho_i^n - \rho_{i-1}^n}{\Delta x} = 0 \tag{8}$$

$$\rho_i^{n+1} = \rho_i^n + \frac{\partial \rho}{\partial t}\Delta t + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t^2 + O(\Delta t^3) \tag{9}$$

$$\rho_{i-1}^n = \rho_i^n - \frac{\partial \rho}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x^2 + O(\Delta x^3) \tag{10}$$

The lengthy equation 11 can be reduced to equation 12 since the ρ_i^n terms subtract out and the Δt and Δx terms in the denominator cancel out. This reduced equation can the be re-written into equation 13, with the original PDE followed by the truncation terms. Notice how the terms on the right are dependent on both the numerical spacing Δt and Δx, but also on the second derivatives of density with respect to space and time.

$$\frac{\left(\rho_i^n + \frac{\partial \rho}{\partial t}\Delta t + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t^2\right) - \rho_i^n}{\Delta t} + U_0 \frac{\rho_i^n - \left(\rho_i^n - \frac{\partial \rho}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x^2\right)}{\Delta x} + O(\Delta x^2, \Delta t^2) = 0 \tag{11}$$

$$\frac{\partial \rho}{\partial t} + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t + U_0\left(\frac{\partial \rho}{\partial x} - \frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x\right) + O(\Delta x^2, \Delta t^2) = 0 \tag{12}$$

$$\frac{\partial \rho}{\partial t} + U_0\frac{\partial \rho}{\partial x} + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t - U_0\frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x + O(\Delta x^2, \Delta t^2) = 0 \tag{13}$$

4 RICHARDSON EXTRAPOLATION

The richardson extrapolation was performed by refining the spatial and temporal step sizes by a factor of 2 for a set number of times. The spatial and temporal studies are refined seperately in their own study in order to isolate the spatial and temporal affects on the the solution. The generation of the inputs, running of the codes, and analysis of the output were automated with a python script in order to reduce user input errors and increase repeatability.

4.1 Convergence of Error

The relative difference at each point for a particular time step was calculated between each iteration for each quantity of interest. For the spatial refinement, the lower iterate values were numerically integrated to match the shape of the initial domain. The errors were then sumned over the entire domain to yield a total error for each variable. The total error for enthalpy is plotted in figures 7 and 8 as a function of temporal and spatial step size respecitvely. The data points where chosen to be outside of the asymptotic range as shown by the good linear fit of the data. The plots show that as the temporal and spatial step sizes are reduced, the numerical error approaches zero.

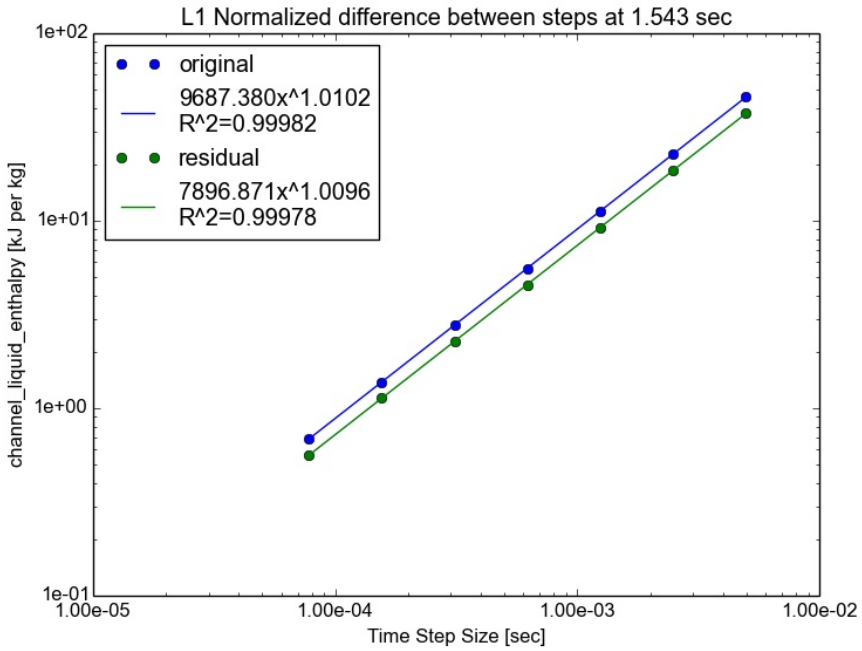

Figure 7. Difference Between Succssessive Temporal Refinements

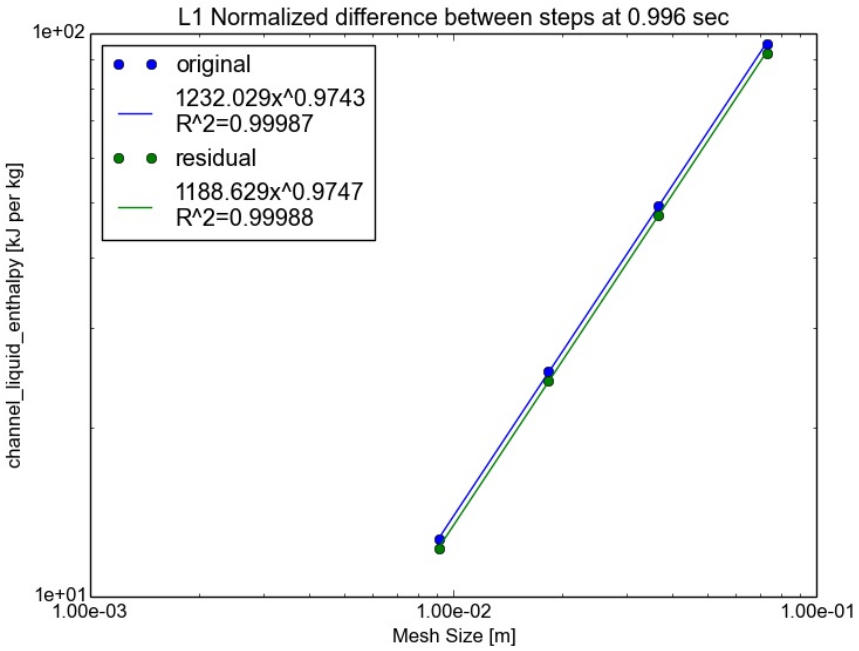

Figure 8. Difference Between Succssessive Temporal Refinements

4.2 Order of Accuracy

The order of accuracy for this verification problem is first order as shown by the modified equation analysis. This can be considred to be the exponent on the power fits as seen in figures 7. However the order of accuracy p can be calculated by using equation 14 where f_1, f_2, f_3 are consecutive lvels within the same Richardson extrapolation study. The refinement factor, R, has the constnat value of 2 for both the spatial and temporal studies.

$$p = \frac{ln\left(\frac{f_3 - f_2}{f_2 - f_1}\right)}{ln(R)} \tag{14}$$

5 CONCLUSIONS

Present your summary and conclusions here.

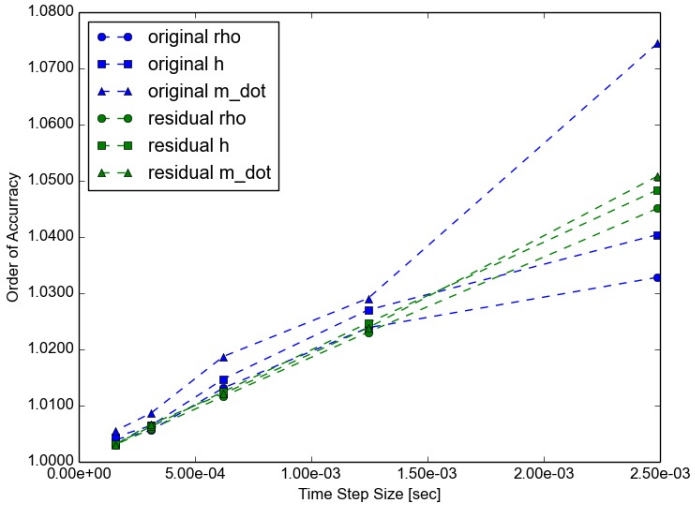

Figure 9. Temporal Order of Accuracy

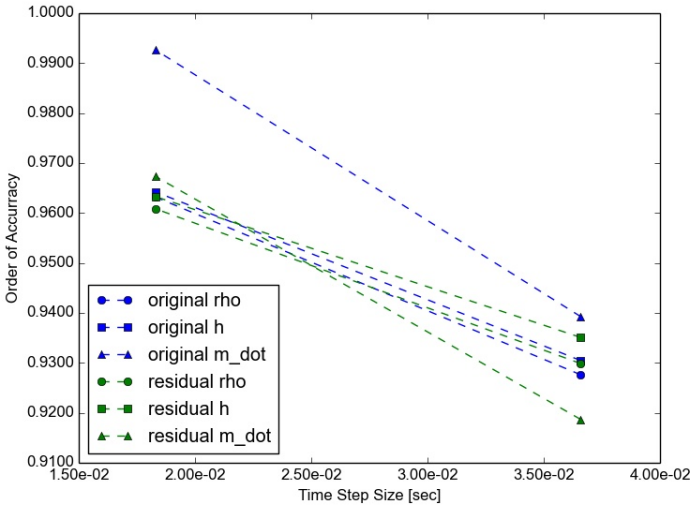

Figure 10. Spatial Order of Accuracy

6 ACKNOWLEDGMENTS

Dr. Vince Mosseau, Dr. Maria Avramova, Dr. Kostadin Ivanov, and Nathan Porter.

7 REFERENCES

[1] R. K. Salko, "CTF Theory Manual," The Pennsylvania State University (2014).

[2] C. J. Roy, "Review of Code and Solution Verification Procedures for Computational Simulation," *J. Comput. Phys.*, **205**, *1*, pp. 131–156 (2005).

[3] K. P. Knupp, *Verification of Computer Codes in Computational Science and Engineering*, Chapman and Hall/CRC, Boca Raton, FL (2003).

[4] IAPWS, "Revised Release on the IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam," The International Association for the Properties of Water and Steam (IAPWS) (2007).