The Pennsylvania State University
The Graduate School
College of Engineering

INITIAL RESIDUAL FORMULATION OF CTF

A Thesis in
Nuclear Engineering
by
Christopher A. Dances

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2015

The thesis of Christopher A. Dances was reviewed and approved* by the following:

Maria Avramova
Associate Professor of Nuclear Engineering
Thesis Advisor, Chair of Committee

Kostadin Ivanov
Distinghuished Professor of Nuclear Engineering

Arthur Motta
Professor of Nuclear Engineering and Materias Science and Engineering
Chair of Nuclear Engineering

Vince Mousseau
Mentor from Sandia National Labs
External Reviewer

*Signatures are on file in the Graduate School.

Abstract

Nuclear engineering codes are being used to simulate more challenging problems and at higher fidelities than they were initially developed for. In order to expand the capabilities of these codes, state of the art numerical methods and computer science need to be implemented. One of the key players in this effort is the Consortium for Advanced Simulation of Light Water Reactors (CASL) and through development of the Virtual Environment for Reactor Applications (VERA). The sub-channel thermal hydraulic code used in VERA, COBRA-TF (Coolant-Boiling in Rod Arrays - Three Fluids), is partially developed at the Pennsylvania State University by the Reactor Dynamics and Fuel Management Research Group (RDFMG).

Currently, 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. The conservation equations analytically reduce into a pressure matrix and are solved using a semi-implicit method. The solid conduction equations are then implicitly solved to determine the temperature within the fuel. Since the liquid solution is solved independent of the solid solution, the solid and liquid equations are explicitly coupled.

In an effort to help meet the objectives of CASL, a version of COBRTA-TF has been developed that solves the residual formulation of the 1D single-phase conservation equations. The formulation of the base equations as residuals allows the code to be run semi-implicitly or fully implicitly while clearly defining the original conservation equations. This paper outlines work to integrate 1D solid conduction equations into the residual formulation. This expands the solid liquid coupling to be either explicit or implicit. Different physical models, such as the homogeneous liquid solid energy model, can be readily implemented by adding the residual functions and variables. A simple test problem consisting of a single liquid channel and fuel pin was designed to compare the original version of COBRA-TF to the different numerical and physical models available through the new residual formulation. The methods are compared both for steady state and transient conditions to quantify the accuracy and stability of each method. The input parameters are varied over a variety of conditions to demonstrate when different methods are most appropriate. The ability to choose appropriate numerical methods and physical models will allow for greater fidelity, decrease computational expenses.

Table of Contents

List of Figures

List of Tables

List of Symbols

ρ Density

h Enthalpy

u Velocity

P Pressure

t Time

g Gravitional Acceleration

Acknowledgments

The Constortium for the Advanced Simulation of Light Water Reactors (CASL) The Reactor Dynamics Fuel Management Group (RDFMG) at Penn State The Toshiba Westinghouse Fellows Program

Dedication

To my parents, and everyone who has supported me in this.

Chapter 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]. The conservation equations analytically reduce into a pressure matrix in a semi-implicit method with rod temperatures solved for explicitly. Because the physics are integrated into the numerical solution, the equations must be linear and the solution method semi-implicit. With a residual formulation, greater flexibility and control over the numerical solution is possible. COBRA-TF was originally written in FORTRAN 77, but over the years has been partially updated to newer versions of Fortran.

Chapter 2 | The Euler Equations

The finite volume structure in COBRA-TF in figure 2.1 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 three dimensional, with channels connecting to each other in two more dimmensions. Fully 3-D equations will be considered in future work.

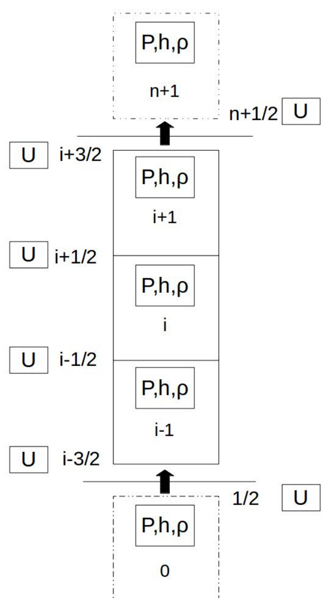

Figure 2.1. The finite volume structure for COBRA-TF

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. This work involves representing the 1-D single phase liquid conservation equations and calculated variables in a residual formulation. The full jacobian matrix can then be built numerically, and can then either be reduced to a pressure matrix or solved directly. Verification of the residuals was done by comparing calculated results to analytical solutions for isokinetic advection and shock tube problems. For each verification problem, a scaling study of the truncation error was compared to the predicted behaviour derived from modified equation analysis using Richardson extrapolation. Further work was then applied to represent 1-D heat conduction within the heater rods. Some initial work was done to allow the code to solve either semi-implicitly, or fully impicitly. The single phase Euler partial differential equations for mass (2.1), momentum (2.2), and energy (2.3) 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.

$$\frac{\partial \rho}{\partial t} + \bigtriangledown \rho u = 0 \tag{2.1}$$

$$\frac{\partial \rho u}{\partial t} + \bigtriangledown \rho u^2 + \bigtriangledown P - \rho g = 0 \tag{2.2}$$

$$\frac{\partial \rho h}{\partial t} - \frac{\partial P}{\partial t} + \bigtriangledown (\rho u h) = 0 \tag{2.3}$$

The 1-D formulation of the Euler Equations will assume a direction x as shown in the 1-D mass equation (2.4). The momentum and energy equations are represented in a non-conservative form as shown in equations (2.5) and (2.7). The momentum equation contains a term that has a product of the left hand side of the 1-D mass equation. This terms can therefore be dropped since it is equivalent to zero, and the entire equation can be divided by density to give a simpler form of the momentum equation (2.6).

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} = 0 \tag{2.4}$$

$$\rho \frac{\partial u}{\partial t} + u \left(\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} \right) + \rho u \frac{\partial u}{\partial x} + \frac{\partial P}{\partial x} - \rho g = 0 \tag{2.5}$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + \frac{1}{\rho} \frac{\partial P}{\partial x} - g = 0 \tag{2.6}$$

$$\rho \frac{\partial h}{\partial t} - \frac{\partial P}{\partial t} + h \frac{\partial \rho}{\partial t} + \rho u \frac{\partial h}{\partial x} + h \frac{\partial \rho u}{\partial x} = 0 \tag{2.7}$$

The 1-D equations are then evaluated at a position index i and a certian time n in order to solve for the next time value of $n+1$. In the mass equation (2.8), the velocities are located at the cell faces $i + \frac{1}{2}$ and $i - \frac{1}{2}$. The density at a corresponding face is either upwinded $\dot{\rho}^n_{i+\frac{1}{2}}$, or averaged $\bar{\rho}^n_{i+\frac{1}{2}}$. In equation (2.9), the derivative $\frac{\partial u}{\partial x}$ is upwinded assuming that the flow is positive. In the energy equation, (2.10) the enthalpy values in the first spatial term are upwinded and shown here assuming a positive velocity. The equation of state (2.11) solves for density assuming that it is a linear combination of changes due to pressure and enthalpy. The partial derivatives in the equation are calculated from steam tables as functions of old time pressure and enthalpy.

$$\frac{\rho_i^{n+1} - \rho_i^n}{\Delta t} + \frac{\dot{\rho}^n_{i+\frac{1}{2}} u^{n+1}_{i+\frac{1}{2}} - \dot{\rho}^n_{i-\frac{1}{2}} u^{n+1}_{i-\frac{1}{2}}}{\Delta x} = 0 \tag{2.8}$$

$$\frac{u^{n+1}_{i+\frac{1}{2}} - u^n_{i+\frac{1}{2}}}{\Delta t} + u^n_{i+\frac{1}{2}} \left(\frac{u^n_{i+\frac{1}{2}} - u^n_{i-\frac{1}{2}}}{\Delta x} \right) + \frac{1}{\bar{\rho}^n_{i+\frac{1}{2}}} \frac{P^{n+1}_{i+1} - P^{n+1}_i}{\Delta x} - g = 0 \tag{2.9}$$

$$\rho_i^n \frac{h_i^{n+1} - h_i^n}{\Delta t} + h_i^n \frac{\rho_i^{n+1} - \rho_i^n}{\Delta t} - \frac{P_i^{n+1} - P_i^n}{\Delta t} + (\rho u)_i^n \frac{h_i^n - h_{i-1}^n}{\Delta x} + h_i^n \frac{\dot{\rho}^n_{i+\frac{1}{2}} u^{n+1}_{i+\frac{1}{2}} - \dot{\rho}^n_{i-\frac{1}{2}} u^{n+1}_{i-\frac{1}{2}}}{\Delta x} = 0 \tag{2.10}$$

$$\rho_i^{n+1} - \rho_i^n = \left(\frac{\partial \rho}{\partial P} \right) \left(P_i^{n+1} - P_i^n \right) + \left(\frac{\partial \rho}{\partial h} \right) \left(h_i^{n+1} - h_i^n \right) \tag{2.11}$$

Chapter 3
Residual Formulation

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 as shown in equations (3.1), (3.2), (3.3), and (3.4). Residuals can also be applied to the conservation equations by substituting the definition of the residual variables into the conservation equations. This will effectively change any variables evaluated at $n + 1$ to 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 as seen in equation (3.5).

$$\delta P_i = P_i^{n+1} - P_i^k \tag{3.1}$$

$$\delta h_i = h_i^{n+1} - h_i^k \tag{3.2}$$

$$\delta u_{i+\frac{1}{2}} = u_{i+\frac{1}{2}}^{n+1} - u_{i+\frac{1}{2}}^k \tag{3.3}$$

$$\delta \rho_i = \rho_i^{n+1} - \rho_i^k \tag{3.4}$$

$$J \delta X = -F(X) \tag{3.5}$$

The Jacobian matrix is defined in equation (3.6) 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 (3.7) 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.

$$J_{i,j} = \frac{\partial F_j(X)}{\partial X_i} \tag{3.6}$$

$$J_{i,j} \approx \frac{F_j(X_i + \epsilon) - F_j(X)}{\epsilon} \tag{3.7}$$

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 3.1.

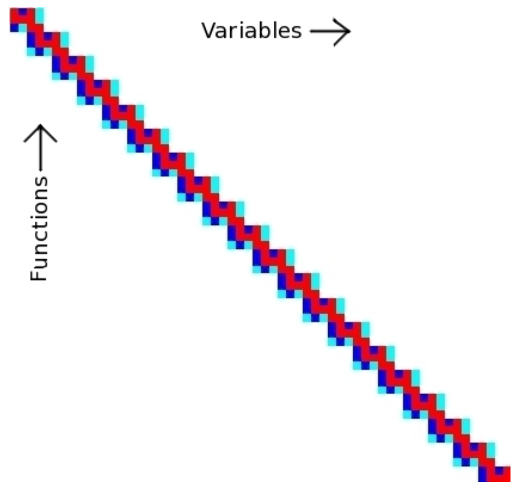

Figure 3.1. Strucutre of the jacobian matrix for single phase liquid

Chapter 4 | Isokinetic Sine Wave Advection

Code verification is the set of procedures set in place to ensure that the code was written properly. From least to most rigorous, the procedures are expert judgement, error quantification, consistency / convergence, and order of accuracy [2]. For this work, the Richardson Extrapolation will be used to check for convergence and order of accuracy of the error in space and time. The error should converge to zero, and the order of accuracy should converge to the values obtained through the modified equation analysis at the end of this section.

4.1 Problem Setup

The verification problem is defined as a single horizontal channel with base parameters listed in table 4.1. Channel area and perimeter are constant across the entire length of the channel. No grid spacers are present, and frictional losses are set to zero. Velocity and pressure are assumed to be constant, but small fluctuations may occur due to numerical roundoff. The channel geometry and operating conditions approximate a standard PWR. The inlet of the channel has a constant velocity with a fluctuating enthalpy that corresponds to a standard PWR rod bundle coolant channel.. The length of the transient was defined to be quadruple the time needed for the liquid at the inlet to advect to the outlet.The frequency of the sine wave was defined to generate a full period of a spatial wave across the length of the channel.

The lookup table to vary the inlet enthalpy h and inlet mass flow rate, $\dot{m}$, are genererated from equations 4.1 and 4.2 respectively. The trignometric functions assume constant axial spacing, Δx, and time step size, Δt, where i and j are the spatial and temporal indices. These equations should also behave as the known solutions throughout the entire domain of the problem. The enthalpy and mass flow rate vary proportionally to the density such that an isokinetic boundary condition is created at the inlet. However, this is dependent on the steam tables used in generating the input and calculating the EOS. 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 [3].

Table 4.1. Problem Parameters

Parameter	Symbol	Value	Unit
Axial Length	L	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

$$h(i,j) = \frac{1}{2}\left((h_1 + h_2) + (h_1 - h_2)cos\left(\omega\left(j\Delta t + \frac{i\Delta x}{V_o}\right)\right)\right) \tag{4.1}$$

$$\dot{m}(i,j) = \frac{1}{2}\left((\dot{m}_1 + \dot{m}_2) + (\dot{m}_1 - \dot{m}_2)cos\left(\omega\left(j\Delta t + \frac{i\Delta x}{V_o}\right)\right)\right) \tag{4.2}$$

The comparison between the data table and the output in CTF are shown for enthalpy and mass flow rate in figures 4.1 and 4.2, respectively. The CTF output was read from hdf5 data files at each point in time, which omitted the actual ghost cell where these values were applied. The CTF values are located at the nearest node to the inlet, and will be slightly out of phase to the exact values. For large mesh sizes this small discrepancy is not noticeable.

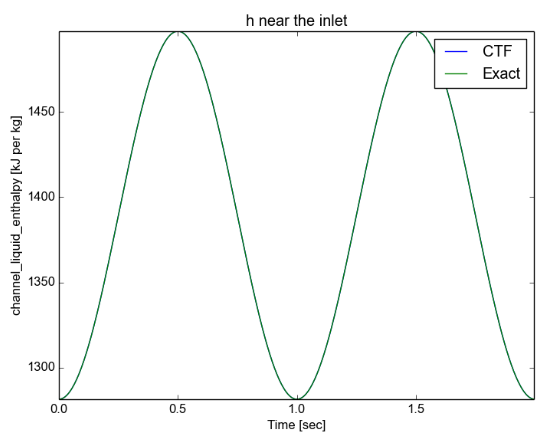

Figure 4.1. Enthalpy Near the Inlet and the Analytical Solution

The pressure and the velocity fluctuate by about 0.5% during the simulation. This is considered small for this problem and should not greatly affect the order of accuracy. The hdf5 output

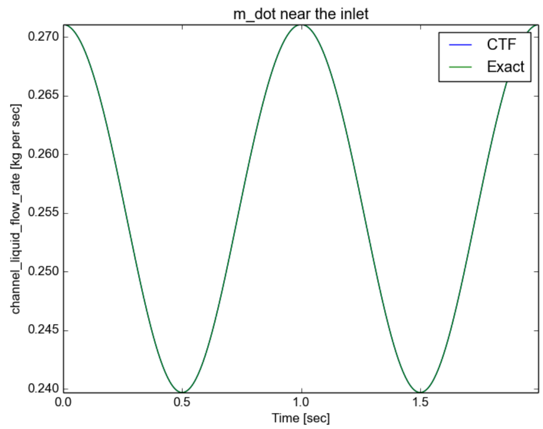

Figure 4.2. Density Near the Inlet and the Analytical Solution

files allow for a high level of precision, reducing round off error in the output during the post processing.

4.2 Code Convergence

The current version of CTF uses global code convergence criteria that are used to estimate the rate of change of global mass and energy conservation. The transient values of these criteria are shown in figure 4.3 for the original version of CTF simulating the verification problem. Mass balance and storage are in units of $\frac{kg}{s}$. The energy balance, fluid energy, and solid energy are in units of kW. The solid energy storage is zero since there are not any heat structures present. The fluctuating values represent differences between the energy and mass entering and leaving the system. The flat profile for the mass storage term means that the sine wave has fully developed spatially through the channel.

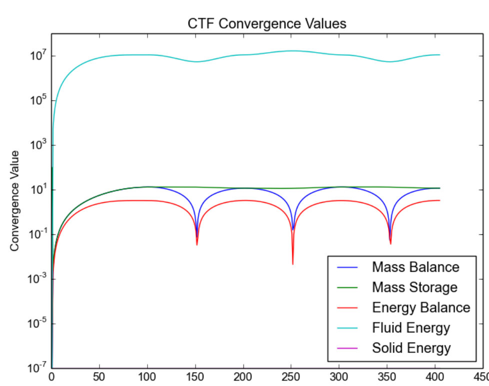

Figure 4.3. Code Convergence Criteria for the Original Version of CTF

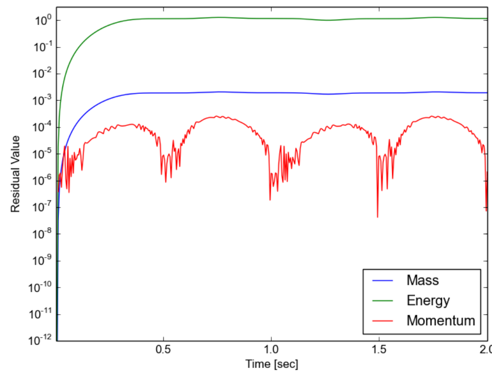

Figure 4.4. Summation of the Residuals for the Residual Version of CTF

The residual formulation prints out the summation of the equation residuals across the domain to an output file at the end of each time step and can be seen in figure 4.4. The mass equation residual is in units of $\frac{kg}{m^3 s}$. The energy equation residual is in units of $\frac{kW}{m^3}$. The momentum residual is in units of $\frac{kg}{m^2 s^2}$. The flat profile of the mass and energy residuals shows that the sine wave has fully developed spatially through the channel.

4.3 Modified Equation Analysis

The order of accuracy in time and space can be analytically determined for this problem through a modified equation analysis. Because the velocity is constant, it can be pulled out of the spatial derivative as shown in equation 4.3. Using upwinding, the finite difference can be written to look like equation 4.4. A second order Taylor series approximation can be used for ρ_i^{n+1} and ρ_{i-1}^n as shown in equations 4.5 and 4.6 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 4.4 to yield 4.7. This is the beginning of the modified equation analysis. The goal will be to isolate the original PDE and define the truncation error.

$$\frac{\partial \rho}{\partial t} + U_0 \frac{\partial \rho}{\partial x} = 0 \tag{4.3}$$

$$\frac{\rho_i^{n+1} - \rho_i^n}{\Delta t} + U_0 \frac{\rho_i^n - \rho_{i-1}^n}{\Delta x} = 0 \tag{4.4}$$

$$\rho_i^{n+1} = \rho_i^n + \frac{\partial \rho}{\partial t}\Delta t + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t^2 + O(\Delta t^3) \tag{4.5}$$

$$\rho_{i-1}^n = \rho_i^n - \frac{\partial \rho}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x^2 + O(\Delta x^3) \tag{4.6}$$

The lengthy equation 4.7 can be reduced to equation 4.8 since the ρ_i^n terms subtract out and the Δt and Δx terms in the denominator cancel out. This reduced equation can be re-written into equation 4.9, with the original PDE followed by the truncation terms.

$$\frac{\left(\rho_i^n + \frac{\partial \rho}{\partial t}\Delta t + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t^2\right) - \rho_i^n}{\Delta t} + U_0 \frac{\rho_i^n - \left(\rho_i^n - \frac{\partial \rho}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x^2\right)}{\Delta x} + O(\Delta x^2, \Delta t^2) = 0 \quad (4.7)$$

$$\frac{\partial \rho}{\partial t} + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t + U_0\left(\frac{\partial \rho}{\partial x} - \frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x\right) + O(\Delta x^2, \Delta t^2) = 0 \quad (4.8)$$

The terms to the right of the original PDE are the first order accurate truncation terms. Notice how the truncation error is dependent on both the on the second derivatives of density with respect to space and time, and on the numerical spacing Δt and Δx. Since the truncation error is linearly dependent on Δt and Δx, the order of accuracy is 1 with respect to time and space.

$$\frac{\partial \rho}{\partial t} + U_0\frac{\partial \rho}{\partial x} + \frac{1}{2}\frac{\partial^2 \rho}{\partial t^2}\Delta t - U_0\frac{1}{2}\frac{\partial^2 \rho}{\partial x^2}\Delta x + O(\Delta x^2, \Delta t^2) = 0 \quad (4.9)$$

When the energy equation undergoes a similar modified eqution analysis, the order of accuracy is also 1 for time and space. The momentum conservation equation does not apply for this problem since the velocity is constant.

4.4 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 separately in their own study in order to isolate the spatial and temporal affects on 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. For this analysis, a significant amount of information was added to the hdf5 output files, increasing memory usage and run time. The computational resources for the spatial study was much higher than the temporal study due to the need to keep the courant number below 0.500. To keep the computational resources needed to perform this analysis reasonable, fewer spatial refinements were performed compared to the temporal analysis.

4.5 Convergence of Error

The difference between iterations was computed at each time step and spatial location for each quanitity of interest. This difference is considered as the error between each iteration. For the spatial refinement, the lower iterate values were numerically integrated to match the shape of the initial domain. The errors were then summed over the entire domain to yield a total error for each variable. The total error for density is plotted in figures 4.5 and 4.6 as a function of temporal and spatial step size.

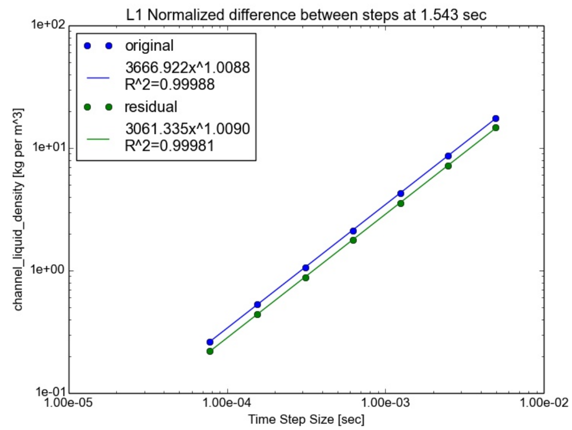

Figure 4.5. Difference Between Successive Temporal Refinements for Density

The data points were chosen to be inside of the asymptotic range as shown by the good power fit with an exponent near 1. The power fit shows that as the temporal and spatial step sizes are reduced, the numerical error approaches zero. The discretization error between the original version of CTF is relatively small and is most likely due to the small fluctuations in the velocity present in the original version of the code.

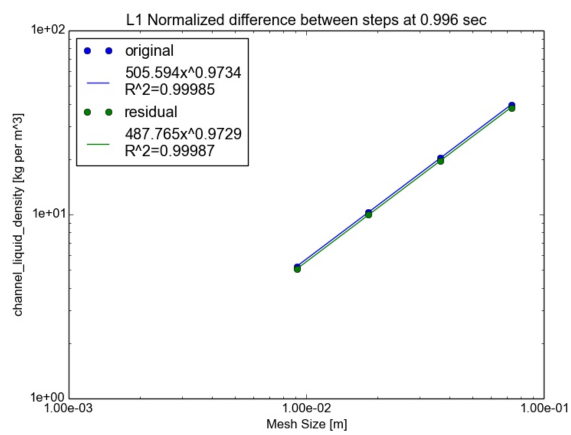

Figure 4.6. Difference Between Successive Spatial Refinements for Density

4.6 Order of Accuracy

The order of accuracy for this verification problem is first order as shown by the modified equation analysis. This can be considered to be the exponent on the power fits as seen in figures 4.5. However the order of accuracy p can be calculated by using equation 4.10 where f_1, f_2, f_3 are consecutive levels within the same Richardson extrapolation study. The refinement factor, R, has the constant value of 2 for both the spatial and temporal studies.

$$p = \frac{ln\left(\frac{f_3 - f_2}{f_2 - f_1}\right)}{ln(R)} \tag{4.10}$$

The order of accuracy for all of the variables are presented for the temporal analysis and spatial analysis in figures 4.7 and 4.8 respectively. The temporal order of accuracy is well within the asymptotic range for the whole analysis, and moves closer to 1.0 with decreasing time step size. The spatial order of accuracy is a slightly outside the asymptotic range, but approaches an order of accuracy of 1.0 with decreasing mesh size.

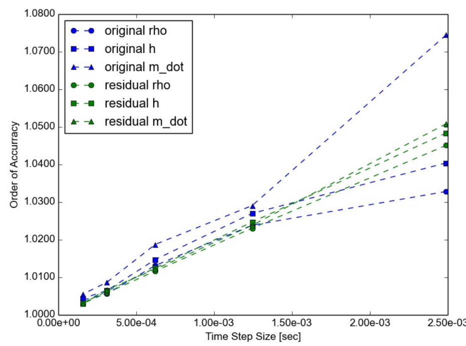

Figure 4.7. Temporal Order of Accuracy

The slight differences between the original version of CTF and the residual formulation might be due to the different solution methods and back substitution of variables. Despite the small differences, both versions of the code exhibit order of accuracies very close the values obtained through the modified equation analysis.

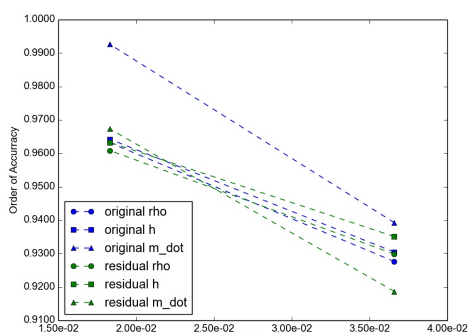

Figure 4.8. Spatial Order of Accuracy

Bibliography

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

[2] Roy, C. J. (2005) "Review of Code and Solution Verification Procedures for Computational Simulation," *J. Comput. Phys.*, **205**(1), pp. 131–156.
URL `http://dx.doi.org/10.1016/j.jcp.2004.10.036`

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