LA-UR-23-29866

Approved for public release; distribution is unlimited.

Title: Verification & Validation of a New Equation of State Framework in xRAGE

Author(s): Gerberding, Seth James
Musick, Benjamin Joel

Intended for: Report

Issued: 2023-08-28

Verification & Validation of a New Equation of State Framework in xRAGE

S.J. Gerberding and B.J. Musick

June 2023

Abstract

This report will detail a the Noh problem as a basic test for Equation of State implementation and verification in hydro codes such as xRAGE. Analytic solutions to a classical test case will be examined for different equations of state and the possibility of obtaining a solution for an arbitrary equation of state will be investigated. The current capabilities of the xRAGE hydrodynamics code will be evaluated against the solutions obtained from the previous analysis. Then, an external equation of state library coupled to xRAGE will be examined and compared both to analytic solutions and the native equation of state unit.

1 Introduction

Motivation for this project lies in verification efforts for the Eulerian xRAGE hydrodynamics code Gittings et al. [2008] and the open-source equation of state library `singularity-eos`. The two primary project objectives are obtaining an analytic solution to a problem that effectively tests hydrodynamic solver properties and expanding Equation of State (EoS) capabilities both from a verification and code capability perspective. The test of interest is the Noh problem Noh [1987], classically run with an ideal gas, to test shock capturing capabilities for a shock moving out from the center of the domain with constant properties in the post-shock (ie, shocked) region. In higher space dimensions, this problem is a good test of a solver's ability to resolve a symmetric interface. A spherical Noh problem slice generated from xRAGE can be seen in Figure 1. An example of a two-dimensional Noh problem from FLASH can be seen in Figure 5, which results in different solutions depending on the hydrodynamic solver employed.

More complicated equations of state allow for simulations that can better resolve realistic, physical systems. Unfortunately, implementation is not always straightforward with both mathematical and numerical challenges. This work will attempt to address verification from an analytic approach where a solution is generated for the Noh problem given an arbitrary EoS and initial conditions. Then the capabilities of the xRAGE hydrodynamics code will be assessed from two different perspectives: xRAGE in its current deployed executable and a version compiled with an external EoS library, `singularity-eos` Peterson et al. [2022][1]. Inspired by the work of Burnett et al. [2017], Yorke et al. [2018], this work will examine the equations of state presented in Burnett et al. [2017] which are the calorically perfect ideal gas law, the stiffened gas law, the Noble-Abel EoS, and the Carnahan-Starling EoS. This work aims to expand both Burnett et al. [2017], which used the FLAG Lagrangian code Crowley [1971] and Yorke et al. [2018] which conducted stiffened gas verification studies in xRAGE.

2 Verification: The Noh problem

One ubiquitous test problem to run is the Noh problem Noh [1987]. One attractive aspect of the problem is the ability to use self-similar variables to take the Euler equations and pose them as an ODE system Axford [2000], Ramsey et al. [2017, 2018]. The question then amounts to solving the ODE system with the Rankine-Hugoniot conditions. Another attractive property of this problem is the ability to test the code in higher dimensions: the problem has a geometrical symmetry (radially symmetric), and thus a code can be tested in one, two, and three dimensions.

The Noh problem has been well-treated in the literature, for instance tc Liska and Wendroff [2003]. However, questions regarding the equation of state and solutions thereof are frequent. Analysis about common equations of state—such as ideal gas, stiff gas, Mie-Gruneisen—is more or less well-understood. However, a more general framework is lacking. Such a framework includes more generic analytic forms of the equation of state as well as so called "black-box" equations of state, such as tabulated Helmholtz free energy or experimental equations of state. Understanding this larger framework requires analyzing constraints on the equation of state that are necessary for an exact solution to exist. The Noh problem is an ideal test problem because, when a solution exists, one

[1] https://lanl.github.io/singularity-eos/main/index.html

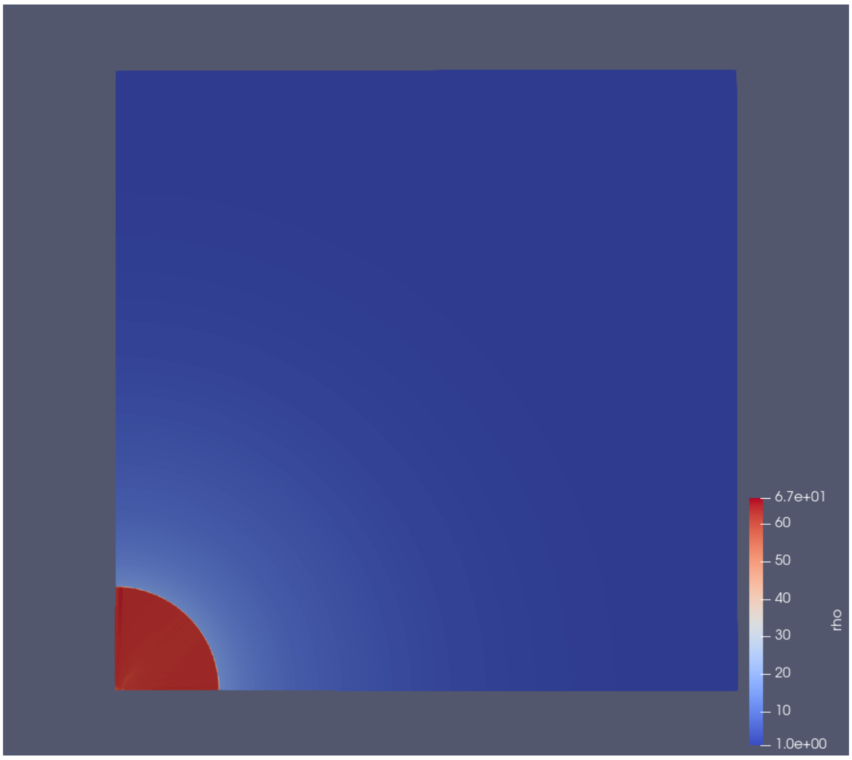

Figure 1: An example of a spherical Noh problem run for a monatomic, calorically perfect gas in the xRAGE hydrodynamics code Gittings et al. [2008].

only needs to know the left variable states: the solution is easily derived from these quantities. However, being able to solve for those variables is not always straightforward.

The Noh Problem consists of a fluid mass of constant density, pressure, and velocity magnitude moving at a constant velocity into a planar wall (in the 1D case), a central axis (in 2D), or a point (in 3D). A shock forms at time $t = 0$ emanating from the symmetry point, leaving behind a fluid mass of constant density and pressure with zero velocity. Analysis shows that the shock wave travels with constant velocity.

We let D denote the shock speed. As noted, $D > 0$ is constant. We are going to let ρ_L, P_L, e_L denote the variable quantities (density, pressure, and internal energy, respectively) behind the shock and ρ_R, P_R, e_R in front of the shock. We also let $m \in \{0, 1, 2\}$ denote the symmetry variable, that is, $m = 0$ corresponds to the Cartesian (planar) case, $m = 1$ the axisymmetric (cylindrical) case, and $m = 2$ the spherically symmetric case. We let r denote the spatial variable and $t \geq 0$ the time variable, and (ρ_0, P_0, u_0) the initial conditions of the problem. Observe that $u_0 < 0$ by assumption of the problem. Finally, we let $\xi := \frac{r}{t}$ denote the self-similar variable and K the adiabatic bulk modulus.

2.1 Solution in the unshocked region

After changing to self-similar variables and conserving mass, momentum and energy, one readily obtains the following solution:

$$\rho_R(x,t) = \rho_0 \left(1 - \frac{u_0}{\xi}\right)^m = \rho_0 \left(1 - \frac{u_0 t}{r}\right)^m \tag{1a}$$

$$P_R = \text{constant} = P_0 \tag{1b}$$

$$u_R = u_0. \tag{1c}$$

An important thing to note, solving the ODE system gives the following equation:

$$\frac{K m u_L}{\xi} = 0.$$

This equation forms a restriction on the equation of state when the geometry is in higher dimensions, which we shall address later. For more details regarading this analysis, we refer the reader to Axford [2000], Ramsey et al. [2017], Burnett et al. [2017].

2.2 Solution in shocked region

In the shocked region, we assume that $u_L = 0$. This leads to the following:

$$\rho_L = \text{constant} \tag{2a}$$

$$P_L = \text{constant} \tag{2b}$$

$$u_L = 0. \tag{2c}$$

Determining the solution becomes a matter of solving for these constants, which is done by solving the Rankine-Hugoniot conditions:

$$(u_L - D)\rho_L = (u_R - D)\rho_R \tag{3a}$$

$$P_L + (u_L - D)\rho_L u_L = P_R + (u_R - D)\rho_R u_R \tag{3b}$$

$$e_L + \frac{P_L}{\rho_L} + \frac{1}{2}(u_L - D)^2 = e_R + \frac{P_R}{\rho_R} + \frac{1}{2}(u_R - D)^2. \tag{3c}$$

Because of the key assumption that $u_L = 0$, the Rankine-Hugoniot conditions simplify to

$$-D\rho_L = (u_0 - D)\rho_R \tag{4a}$$

$$P_L = P_R + (u_R - D)\rho_R u_0 \tag{4b}$$

$$e_L + \frac{P_L}{\rho_L} + \frac{1}{2}D^2 = e_R + \frac{P_R}{\rho_R} + \frac{1}{2}(u_0 - D)^2 \tag{4c}$$

Simplifying the last equation gives the final form of the Rankine-Hugoniot conditions,

$$-D\rho_L = (u_0 - D)\rho_R \tag{5a}$$

$$P_L = P_R + (u_R - D)\rho_R u_0 \tag{5b}$$

$$e_L + \frac{P_L}{\rho_L} = e_R + \frac{P_R}{\rho_R} + \frac{1}{2}u_0(u_0 - 2D) \tag{5c}$$

Using these Rankine-Hugoniot conditions with Equation (1a) for the unshocked density, we have explicit expressions for the left state:

Lemma 2.1. *Consider the Noh problem and assume a solution exists. Then the left density and pressure states are given by:*

$$\rho_L = \frac{D - u_0}{D}\rho_R = \rho_0 \left(\frac{D - u_0}{D}\right)^{m+1} \tag{6a}$$

$$P_L = P_R + (u_0 - D)u_0 \left(\frac{D - u_0}{D}\right)^m \rho_0 \tag{6b}$$

Proof. Equation (6a) follows from (5a) and dividing by $-D$. Equation (6b) follows from the fact that ρ_R is (1a) evaluated at $r/t = D$. □

The advantage of this formulation is that, once the shock speed D is solved for, one can easily compute the left and pressure state. As we shall see, the key to solve the Noh problem is solving the left density state, then the shock speed, and finally the left pressure state.

2.3 Constraints on the equation of state

As Ramsey et al. [2017] points out, the geometry of the problem greatly influences the admissible EoSs. In particular, in order for a solution to exist, we need $K = 0$ or $m = 0$. When $m \neq 0$, $K = 0$ implies that, in the unshocked region, the following must hold at all positions and times:

$$e + \frac{P_0}{\rho_0}\left(1 - \frac{u_0 t}{r}\right)^{-m} = \text{constant}. \tag{7}$$

That is to say, the EoS must take the form:

$$e(\rho(t,r), P(t,r)) = C - \frac{P_0}{\rho_0}\left(1 - \frac{u_0 t}{r}\right)^{-m}. \tag{8}$$

While this is possible in the abstract, it's going to exclude many equations of state, at least in the higher symmetries. Indeed, in practice the only EoSs that will hold in higher symmetries are ones where $P_0 = 0$ implies that $e(\rho, P_0) = \text{constant}$.

To summarize: when $m = 0$, the 1D planar case, most EoSs can be used under very mild assumptions (for example, that it can posed as $e = e(\rho, P)$). However, when $m \neq 0$, many equations of state will preclude analytic solutions from existing. Therefore, for verification purposes, one can hope to test using the Noh problem in the planar geometry (one still has to find the exact solution). However, in higher symmetries, depending on the EoS, analytic solutions may simply not exist.

2.4 Solutions for a class of equations of state

In this section, we present mathematical results regarding the exact solution to the Noh problem when the equation of state is in a particular generic form. One advantage is that this generic form covers equations of state ranging from ideal gas to Noble-Abel. That is, we give the solution for a wide class of equations of state and all admissible initial conditions and symmetries and a method for solving them. Here, we are agnostic with respect to the symmetry of the problem, but we do assume that the initial pressure is zero, ie, $P_0 = 0$. This is not an outrageous assumption—indeed, when $m \neq 0$, $P_0 = 0$ is frequently a requirement for a solution to exist. For practical purposes, the only case when $P_0 \neq 0$ is when $m = 0$. We do not address this case—it is an area worthy of future research.

For the analysis, we suppose that the equation of state is given in form:

$$e(\rho, P) = \frac{P}{\rho}\beta(\rho), \qquad \beta(\rho) > 0. \tag{9}$$

With this assumption, we are able to know what the shock speed will be:

Lemma 2.2. *Consider the Noh Problem with $P_0 = 0$ and suppose the equation of state takes the form in* (9)*. Then the shock speed is given by*

$$D = -\frac{u_0}{2\beta(\rho_L)} \tag{10}$$

Proof. Since $u_L = 0$ by assumption the Rankine-Hugoniot conditions become

$$\rho_L = \frac{D - u_R}{D}\rho_R \tag{11a}$$

$$P_L = (u_R - D)u_R\rho_R \tag{11b}$$

$$e_L + \frac{P_L}{\rho_L} = \frac{1}{2}u_R(u_R - 2D) = \frac{1}{2}u_0(u_0 - 2D). \tag{11c}$$

Divide (11b) by (11a) to obtain

$$\frac{P_L}{\rho_L} = -Du_0. \tag{12}$$

Thus (11c) simplifies to

$$e_L = \frac{1}{2}u_0^2. \tag{13}$$

Now we substitute (9) in for e_L, and so we have

$$\frac{P_L}{\rho_L}\beta(\rho_L) = -Du_0\beta(\rho_L) = \frac{1}{2}u_0^2 \tag{14}$$

and therefore

$$D = -\frac{1}{2\beta(\rho_L)}u_0 \tag{15}$$

□

We can use this fact to find am implicit solution for the left density state.

Lemma 2.3. *Consider the Noh Problem with $P_0 = 0$ and suppose the equation of state takes the form in* (9)*. Then the left density state satisfies:*

$$\rho_L^{\frac{1}{m+1}} - 2\beta(\rho_L)\rho_0^{\frac{1}{m+1}} = \rho_0^{\frac{1}{m+1}} \tag{16}$$

Proof. Since $u_L = 0$, (3a) becomes

$$\rho_L = \frac{D - u_0}{D}\rho_R. \tag{17}$$

We can use (1a) at the shock to get the right density state

$$\rho_R = \rho_0 \left(\frac{D-u_0}{D}\right)^m = \rho_0 \left(\frac{-\frac{1}{2\beta(\rho_L)}u_0 - u_0}{-\frac{1}{2\beta(\rho_L)}u_0}\right)^m = \rho_0 \left(\frac{\frac{1}{2\beta(\rho_L)}+1}{\frac{1}{2\beta(\rho_L)}}\right)^m = \rho_0 \left(1 + 2\beta(\rho_L)\right)^m. \tag{18}$$

Thus (17) becomes

$$\rho_L = \rho_0 \left(1 + 2\beta(\rho_L)\right)^{m+1}. \tag{19}$$

Manipulating this equation gives the result.

□

Finally, we can give the full solution to the Noh problem with this form of the equation of state.

Theorem 2.4. *Let ρ^* be the solution to* (16)*. Assume $\beta(\rho^*) > 0$. Then the solution to the Noh Problem is given by* (1a) - (1c) *and*

$$\rho_L = \rho^* \tag{20a}$$

$$P_L = \rho_0 u_0^2 \frac{(1+2\beta(\rho^*))^{m+1}}{2\beta(\rho^*)} \tag{20b}$$

Proof. Equation (20a) follows from the lemma, and (20b) follows from (6b) and the value for D found in the first lemma. □

As an example, we can use this theory to give analytical solutions to the Noh Problem using the Noble-Abel equation of state, which we assume to take the form:

$$e(\rho, P) := \frac{P}{\rho}(a\rho + b); \quad a < 0,\ b > 0,\ a\rho + b > 0. \tag{21}$$

In terms of the theory, we have $\beta(\rho) = a\rho + b > 0$. The constraint to be solved for is therefore:

$$\rho_L^{\frac{1}{m+1}} - 2a\rho_L\rho_0^{\frac{1}{m+1}} = 2b\rho_0^{\frac{1}{m+1}} + \rho_0^{\frac{1}{m+1}} \tag{22}$$

One convenient fact about this EoS is it can always be easily solved, which we summarize and prove here:

Proposition 2.1. *There exists a unique positive solution for ρ_L to* (22)*.*

Proof. Since $a < 0$, (16) takes the form

$$x^{\frac{1}{m+1}} + Kx - C =: f(x) \tag{23}$$

with $K, C > 0$. Clearly, $f(0) = -C < 0$. Now $f'(x) = \frac{1}{m+1}x^{\frac{-m}{m+1}} + K$. In all cases, $m = 0$, $m = 1$ and $m = 2$ $f'(x) > 0$ for all $x \neq 0$. Thus, by the intermediate value theorem and mean value theorem, there exists a unique $x > 0$ such that $f(x) = 0$. □

We can now summarize the analysis for the Noh Problem when the Noble-Abel equation of state is used:

Corollary 2.1. *The solution to the Noh Problem using the Noble-Abel equation of state. Let ρ^* solve* (22) *and that $\rho^* < \frac{-b}{a}$. Then the solution to the Noh problem with $P_0 = 0$ is given by:*

$$\rho_L = \rho^* \tag{24a}$$

$$P_L = \rho_0 u_0^2 \frac{(1 + 2a\rho^* + 2b)^{m+1}}{2a\rho^* + 2b} \tag{24b}$$

Proof. Follows from the theorem. □

Solving (22) can be done algebraically: multiply by ρ_L to get rid of the radical and then solve the (up to quartic) equation. There are formulas for doing so—however, they are rather complicated, and even if one does push through the computations, they have to shift through the roots and decide which one is the correct choice. Alternatively, easy analysis of (16) shows that a Newton solver can easily arrive at the solution with initial condition close to zero. Indeed, we recommend this approach.

2.5 Black Box equation of state

A question that has yet to be addressed is how to verify codes when the equation of state is black box, that is, we have a mapping $e(\rho, P)$, but we don't know the expression. Examples of such equations of state include tabulated equations of state such as SESAME tables: the user gives the table density and pressure (for example) and receives the energy. Other assumptions can be made, for example, you can also ask the EOS about derivative values, ie, $\frac{\partial e}{\partial \rho}$, etc.

When the EoS is a black box, deriving an analytic solution may very well *not* be possible, and so how to verify a code is an open problem. However, because of the nature of the Noh problem, the only thing that needs to be known is the left variable states and the shock speed. This amounts to being able to solve the Rankine-Hugoniot conditions. However, this is not a simple question. There are a few assumptions that make things easier.

Lemma 2.5. *If $P_0 = P_R = 0$, then the following relation holds*

$$\frac{P_L}{\rho_L} = -Du_0 \tag{25}$$

Proof. We have the following Rankine-Hugoniot conditions

$$\rho_L = \frac{D - u_0}{D} \rho_R \tag{26a}$$

$$P_L = (u_0 - D)\rho_R u_0 \tag{26b}$$

Thus

$$\frac{P_L}{\rho_L} = \frac{(u_0 - D)\rho_R u_0}{\frac{D-u_0}{D}\rho_R} = -Du_0 \tag{27}$$

□

Of course, when the EOS is a black box, the classical notion of an analytic solution is no longer available. That is to say, it is impossible to write a closed form of the solution as we did in the previous section. From a verification perspective, this leaves open the question of how to test a code. Fortunately, the Noh problem is such that we only need to solve the Rankine-Hugoniot conditions—this will give us the shock speed D and the shock-region values. Once these are known, we are able to reconstruct an exact solution.

One possibility for solving the Rankine-Hugoniot conditions involves inverting a 3x3 matrix system. However, we can make things a little simpler by using the assumptions of the Noh problem. In particular, we have the following:

Lemma 2.6. *For the Noh problem in the 1D planar case with $P_0 = 0$, the Rankine-Hugoniot conditions can be simplified to the following:*

$$P_L - \frac{P_L \rho_0}{\rho_L} = u_0^2 \rho_0 \tag{28a}$$

$$e(\rho_L, P_L) = e(\rho_R, P_R) + \frac{P_R}{\rho_R} + \frac{1}{2}u_0^2. \tag{28b}$$

Proof. Recall the following conditions:

$$e_R + \frac{P_L}{\rho_L} = e_R + \frac{1}{2}u_0(u_0 - 2D) \tag{29a}$$

$$P_L = (u_0 - D)\rho_R u_0 = (u_0 - D)\rho_0 u_0 \tag{29b}$$

Using that $\frac{P_L}{\rho_L} = -Du_0$, we have:

$$e_L - Du_0 = e_R + \frac{1}{2}u_0(u_0 - 2D) = e_R + \frac{1}{2}u_0^2 - Du_0. \tag{30}$$

$$P_L = (u_0 - D)\rho_0 u_0 = \rho_0 u_0^2 + \frac{P_L}{\rho_L}\rho_0 \tag{31}$$

Rearranging these equations gives the result. □

Our approach then is to solve for ρ_L and P_L at the same time using a vector-valued Newton solver. That is, we define

$$F(\rho, P) := \begin{pmatrix} P - \frac{P\rho_0}{\rho} - u_0^2\rho_0 \\ e(\rho, P) - e(\rho_R, P_R) - \frac{P_R}{\rho_R} - \frac{1}{2}u_0^2 \end{pmatrix}. \tag{32}$$

And so we are seeking ρ, P such that $F(\rho, P) = 0$. If let $DF(\rho, P)$ denote the Jacobian matrix of F evaluated at ρ, P, the Newton solver includes an initial guess $\mathbf{x}_0$ and the following iteration (which can be written two ways):

$$\mathbf{x}_{n+1} = \mathbf{x}_n - (\mathbf{DF})^{-1}(\mathbf{x}_n)\mathbf{F}(\mathbf{x}_n) \quad \text{(Explicit)} \tag{33}$$

$$(\mathbf{DF})(\mathbf{x}_n)\mathbf{x}_{n+1} = (\mathbf{DF})(\mathbf{x}_n)\mathbf{x}_n - \mathbf{F}(\mathbf{x}_n) \quad \text{(Implicit solve)} \tag{34}$$

We measure the residual, which by construction, is simply

$$\|\mathbf{F}(\mathbf{x}_{n+1})\| \tag{35}$$

and we iterate until we are within some tolerance.

Now, a key question is whether or not the matrix **DF** is invertible given a point (ρ, P). Since **DF** is a 2x2 matrix, checking the determinant is easy, and we want it to be nonzero. In short, we require:

$$\det(\mathbf{DF}(\rho, P)) = \frac{\partial e}{\partial P}(\rho, P)\frac{P}{\rho^2}\rho_0 - \left(1 - \frac{\rho_0}{\rho}\right)\frac{\partial e}{\partial \rho}(\rho, P) \neq 0 \tag{36}$$

Since the EOS is a black box, there is not much more we can say here *a priori*, save for a sign analysis, which we give here:

Proposition 2.2. *Suppose $\rho > \rho_0$. If sign$(\frac{\partial e}{\partial \rho}) = -$sign$(\frac{\partial e}{\partial P}P)$ (pointwise save for $\rho = 0$) then the Jacobian matrix is always invertible.*

Proof. Follows from (36) and the fact that nonzero determinant is equivalent to a matrix being invertible. □

While this is not a necessary condition, it provides an easy sufficient condition to check. Furthermore, other thermodynamic considerations may be leveraged to further prove invertibility. One area of research is a question of invariance; that is to say, if (ρ_n, P_n) exist in a certain space, is (ρ_{n+1}, P_{n+1}) also in that space? If a property like this holds, there's hope of proving convergence of the method as long as the initial guess is chosen in a smart way.

3 Software

Primary software contributions from this work are in the form of an analytic solution library and additional equations of state in the `singularity-eos` library. The solution library is currently capable of providing solutions to the Noh problem for select equations of state. Equations of state added to Singularity were those utilized in Burnett et al. [2017].

3.1 Analytic Solution Library

The analytic solution library consists of a single file `exact_solutions.py`, which contains a collection of classes. Each of these classes corresponds to an analytic solution of the Noh problem for different EoSs. As of the writing of this report, it includes ideal gas, stiff gas, and Noble-Abel solutions. More solutions ought to be added as they are derived and analyzed. One important thing to note: these solutions assume that the initial pressure is zero, as is reflected in the theory above. More work should be done to derive solutions for when the initial pressure is nonzero. The following describes how the library works.

3.1.1 exact_solutions.py

This script acts as a library of classes that corresponds to analytic solutions for the Noh problem. The intended use for this library is that the user imports it into a separate file and can call the classes as needed.

3.1.2 Initialization, inputs, and member functions

Each class is initiated by giving a dictionary containing a list of variables. Each class requires initial conditions, which includes density, velocity, pressure, and symmetry of the problem. An example would be:

```
conditions = { 'velocity':-1, 'density': 1, 'pressure':0, 'symmetry':0}
```

Note that other classes may have other inputs (such as Noble-Abel and other parameter-based EoSs). However, in those cases, there is a default for the extra inputs. Initializing a solution class is exceptionally simple: most classes are named after the EoS they are based off of, and so calling them is as easy as

```
solution = EoS_name(conditions)
```

3.1.3 Solution

Each class has member functions for evaluating the exact solution at a give time and place. They can be called as follows:

```
solution.density_solution(time,x)
solution.pressure_solution(time,x)
```

3.1.4 Plotting

Each class has member functions for plotting the solution of density and pressure. They permit the user to plot the solution at a given time. They can be called as follows:

```
solution.plot_density(time)
solution.plot_pressure(time)
```

3.1.5 Ideal Gas

Ideal gas only requires the initial conditions and the symmetry of the problem. It can be instantiated as follows:

```
ideal_solution = ideal_gas(conditions)
```

This class is essentially 'hard-coded' insofar as the solution is a formula in terms of the input data and there is no solving required. Note that in higher dimensions, an analytic solution only exists if the initial pressure is 0. The code checks that this condition is met, and if it not, it throws an error.

3.1.6 Stiff Gas

Stiff gas only requires the initial conditions and the symmetry of the problem. It can be instantiated as follows:

```
stiff_solution = stiff_gas(conditions)
```

This class is essentially 'hard-coded' insofar as the solution is a formula in terms of the input data and there is no solving required. The formula can be in several papers. One important thing to note about stiff gas is an analytic solution only exists in 1D planar settings. The class has an assert that guarantees this conditions is met–if it isn't, the code throws an error.

3.1.7 Noble-Abel

Noble-Abel, when instantiated, takes the initial conditions along with EoS parameter value.

```
conditions = { 'velocity':-1, 'density': 1, 'pressure':1, 'symmetry':0}
Noble_solution = Noble_Abel(conditions, b = 0.2)
```

This class is more involved that the others we've outlined. The mathematics, which can be found in this document, show that the solution can be found by solving a constraint equation. (In particular, the constraint gives the left density state, ρ_L). That equation could, in theory, be solved using algebraic methods. However, this can require some obnoxious algebra. Some easy analysis, on the other hand, highly suggests the use of a Newton solver, and that's what we do in this class. Once the equation is solved, the rest of the required variables have formulae in terms of the left density state.

3.1.8 Example

Suppose our initial conditions are in a dictionary with values and we want to use ideal gas in 1D planar. The way to create the exact solution is as follows:

```
import exact_solutions.py as es

conditions = { 'velocity':-1, 'density': 1, 'pressure':1, 'symmetry':0}}
ideal_solution = es.ideal_gas(conditions)
```

3.2 Analysis tool

We also developed an analysis tool that uses the exact solution classes found in `exact_solutions.py`. Briefly, it takes in numerical data and an instance of a solution. The user can plot the two and run L1 error analysis. The details are described below.

3.2.1 analysis.py

This script contains a class that allows the user to input a solution class from `exact_solution.py`, a computed numerical solution, and compare the two. The inputs for class are a solution class from `exact_solution.py`, a list of position or x variables, a list of density values, a list of pressure values, and the number of nodes to plot the analytic solution on, which is defaulted to 500.

3.2.2 Member functions

Member functions of an analysis class are as follows:

- `compute_density_norm()`: computes the L1 norm of the density using the exact solution
- `compute_pressure_norm()`: computes the L1 norm of the pressure using the exact solution
- `compute_L1_density_error()`: computes the L1 error of the density
- `compute_L1_pressure_error()`: computes the L1 error of the pressure
- `plot_density()`: plots the analytical density and the numerical density on the same graph
- `plot_pressure()`: plots the analytical pressure and the numerical pressure on the same graph
- `run()`: executes all of the above functions and returns the values and graphs

3.2.3 Example

Suppose we have a list of x-vales in a list `x_vals`, a list of densities `densities`, and a list of pressure `pressures`. Suppose we are using the ideal gas equation of state. A way to run the analysis would be:

```
import exact_solutions.py as es
import analysis.py as an

conditions = { 'velocity':-1, 'density': 1, 'pressure':1, 'symmetry':0}}
ideal_solution = es.ideal_gas(conditions)
ideal_gas_analysis = an.noh_analysis(ideal_solution, x_vals, densities,
pressures)
ideal_gas_analysis. run()
```

3.3 Equation of State Framework

The `singularity-eos` Peterson et al. [2022] open-source library offers performance portable equations of state and mixed cell closures. Designed with continuum dynamics codes in mind, the library is intended to be easily coupled to high performance codes utilizing either CPU or GPU infrastructures. There are various Equations of State (EoS) implemented in the library, however the focus of this work lies on implementing and verifying the formulations utilized in the work of Burnett et al. [2017]: the Stiffened Gas, Noble-Abel, and Carnahan-Starling EoSs. All equations of state were formulated in the Mie-Gruneisen form as seen in Equation 37.

$$P - P_{ref} = \rho\Gamma(\rho)(e - q) \tag{37}$$

The Stiffened Gas formulation was implemented based on Le Métayer et al. [2004]. The Noble-Abel (Clausius I, Hirn) EoS was implemented using perspective from Neron and Saurel [2022]. The (Quasi-Exact) Carnahan-Starling EoS was derived for a given compressibility factor. The Mie-Gruneisen parameters for each EoS can be seen in Table 1. Extended formulation details can be found in Appendix 5.

Entropy availability was determined through either one or a combination of the relations seen in Equation 38.

$$\begin{aligned}
\Delta S(\rho, T) &= \int_{T_0}^{T} \frac{1}{T}\left(\frac{\partial e}{\partial T}\right)_v dT + \int_{v_0}^{v}\left(\frac{\partial P}{\partial T}\right)_v dv \\
\Delta S(P, T) &= \int_{T_0}^{T} \frac{1}{T}\left(\frac{\partial h}{\partial T}\right)_P - \int_{P_0}^{P}\left(\frac{\partial v}{\partial T}\right)_P \\
\Delta S(\rho, e) &= \int_{e_0}^{e} \frac{1}{T(\rho, e)} de + \int_{v_0}^{v} \frac{P(\rho, e)}{T(\rho, e)} dv
\end{aligned} \tag{38}$$

EoS	$\Gamma(\rho)$	P_{ref}
Stiffened Gas	$\gamma - 1$	$-\gamma P_\infty$
Noble-Abel	$\frac{\gamma-1}{1-b\rho}$	0
Carnahan-Starling	$Z(\gamma - 1)$	0

Table 1: Mie-Gruneisen parameters for the Equations of State presented in Section 3.3. Here, γ is the adiabatic index and b is the covolume of the material. The Carnahan-Starling compressibility factor, $Z(\rho)$, can be found in Appendix 5.

The isentropic bulk modulus was obtained from the relationships seen in Equation 39.

$$\begin{aligned}
\left(\frac{\partial P}{\partial \rho}\right)_S &= \left(\frac{\partial P}{\partial \rho}\right)_e \left(\frac{\partial \rho}{\partial \rho}\right)_S + \left(\frac{\partial P}{\partial e}\right)_\rho \left(\frac{\partial e}{\partial \rho}\right)_S \\
\left(\frac{\partial e}{\partial \rho}\right)_S &= T\left(\frac{\partial S}{\partial \rho}\right)_S - P\left(\frac{\partial v}{\partial \rho}\right)_S \\
\left(\frac{\partial P}{\partial \rho}\right)_S &= \left(\frac{\partial P}{\partial \rho}\right)_e + \frac{P}{\rho^2}\left(\frac{\partial P}{\partial e}\right)_\rho \\
\rho\left(\frac{\partial P}{\partial \rho}\right)_S &= \rho\left(\frac{\partial P}{\partial \rho}\right)_e + P\Gamma
\end{aligned} \tag{39}$$

The implementation of each EoS was done under the assumption of a Centimeter-Gram-Second (CGS) unit system. Utilization of each EoS requires at least the following input from the user: the adiabatic index minus one ($\gamma - 1$), the specific heat at constant volume (C_v), the Stiffened Gas pressure or material covolume (P_∞ or b), and an offset specific internal energy (q). Optional user input allows for the specification of an offset entropy (q') and state (T_0 and P_0). Documentation and source code is available online.[2][3]

4 Simulation Results

Results relevant to the discussion at hand will be laid out in order of progression towards verification methods of equation of state implementations. Iterative solutions to the planar Noh problem will be presented along with a short discussion of EoS implementation into the FLASH4.7 hydrodynamics code. Then, preliminary results from the xRAGE and `singularity-eos` combination will be discussed.

4.1 Planar Noh Iterative Solutions

The Rankine-Hugoniot relationships for a uniform flow area (planar) can be seen in Eqn. 3. Making the assumption that the right state of the shock is at zero pressure and energy, these relationships can be rewritten in the vector valued Newton method as,

$$\mathbf{x} = \begin{pmatrix} \rho_L \\ P_L \text{ or } e_L \end{pmatrix} \tag{40a}$$

$$\mathbf{F} = \begin{pmatrix} P_L(\rho_L - \rho_R) - \rho_L \rho_R u_R^2 \\ e_L + P_L(\frac{1}{\rho_L} - \frac{1}{\rho_R}) + \frac{1}{2}u_R^2 \end{pmatrix} \tag{40b}$$

$$\mathbf{DF} = \begin{bmatrix} \left(\frac{\partial f_1}{\partial \rho_L}\right) & \left(\frac{\partial f_1}{\partial P_L} \text{ or } \frac{\partial f_1}{\partial e_L}\right) \\ \left(\frac{\partial f_2}{\partial \rho_L}\right) & \left(\frac{\partial f_2}{\partial P_L} \text{ or } \frac{\partial f_2}{\partial e_L}\right) \end{bmatrix} \tag{40c}$$

$$D = \frac{\rho_R u_R^2 - P_L}{\rho_R u_R}, \tag{40d}$$

providing the constant post-shock conditions depending on the EoS chosen. The explicit Newton method was employed for the following Noh solutions. It should be noted that the Newton solver is highly sensitive to the initial guess. Derivatives were evaluated analytically during testing for simple constitutive relations but were evaluated using a centered finite difference method in the final implementation for a more general code. Solutions for selected equations of state can be seen in Table 2.

[2]https://lanl.github.io/singularity-eos/main/index.html
[3]https://github.com/lanl/singularity-eos

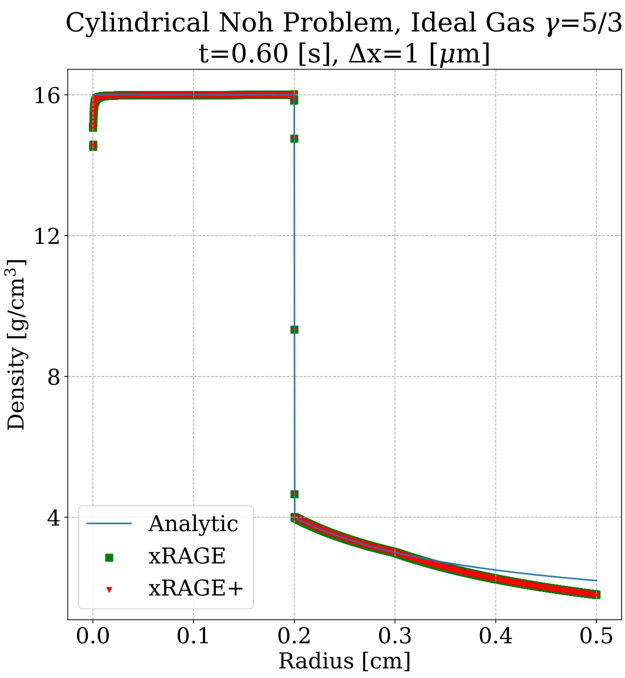

Figure 2: Ideal gas results from xRAGE with and without utilization of the Singularity library. Analytic solution provided for visual comparison.

EoS	P_L	ρ_L
Ideal Gas	$1.\bar{3}$	4
Stiffened Gas	2.119633	1.893150
Noble-Abel	1.346801	3.883495
Carnahan-Starling	1.385820	3.591882
SESAME (He)	1.330626	4.024570
SESAME (Al)	1.189678	6.272099

Table 2: Solutions to the planar Noh problem found from an iterative Newton approach. Ideal/Stiffened Gas, Noble-Abel, and Carnahan-Starling assume $\gamma = 5/3$. The Stiffened gas value of P_∞ was taken as 1 $[dyn/cm^2]$. The covolume was taken to be $b = 0.01$ $[cm^3/g]$ where applicable. Energy offset q was taken as zero for all EoS with the exception of the Stiffened Gas which took $q = -\gamma/(\gamma - 1)$. SESAME Tables were taken from the FLASH4.7 hydrodynamics code that are provided for the `LaserSlab` example problem.

The equations of state relevant to the study at hand were implemented into the FLASH4.7 hydrodynamics code for quick testing. Additionally, two select SESAME equations of state (helium and aluminum) were utilized to test the "black box" functionality of the iterative analytic solution approach for the planar Noh problem. Visual comparisons of analytic solutions and different equation of state implementations in the FLASH code can be see in Appendix 5. Without detailed convergence analysis, the results remain inconclusive, however, visual agreement is obtained to the analytic solutions providing confidence in the chosen solution path.

4.2 xRAGE

As initially stated, the primary objective of this project is to verify the implementation of the Singularity equation of state library into the continuum hydrodynamics code xRAGE. Preliminary results have been obtained for the calorically perfect ideal gas case as seen in Figure 2. Excellent agreement is obtained between normal xRAGE and xRAGE built with `singularity-eos`. Here, there is noticeable disagreement between the analytic solution and the numerical solutions with an unclear root cause. Inflow boundary interactions may be the cause of this discrepancy.

The stiffened gas EoS is currently supported in xRAGE, however like the ideal gas law, this EoS was implemented in Singularity and coupled to xRAGE. Results from a planar Noh test case can be seen in Figure 3

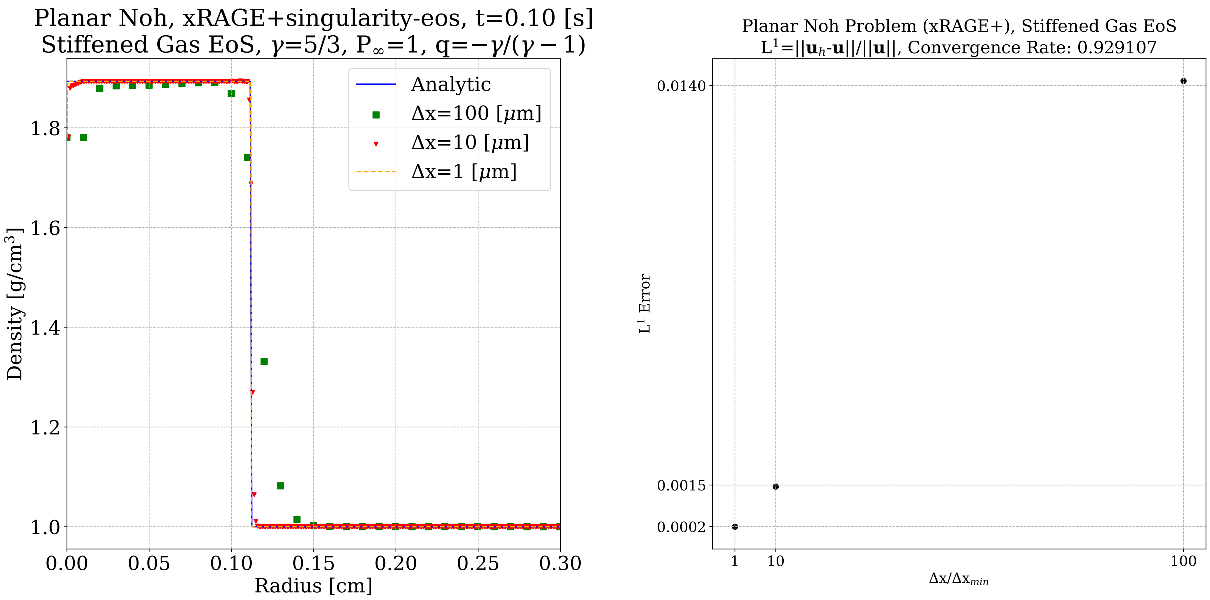

Figure 3: Stiffened gas results from xRAGE with the Singularity library. Analytic solution provided for visual comparison.

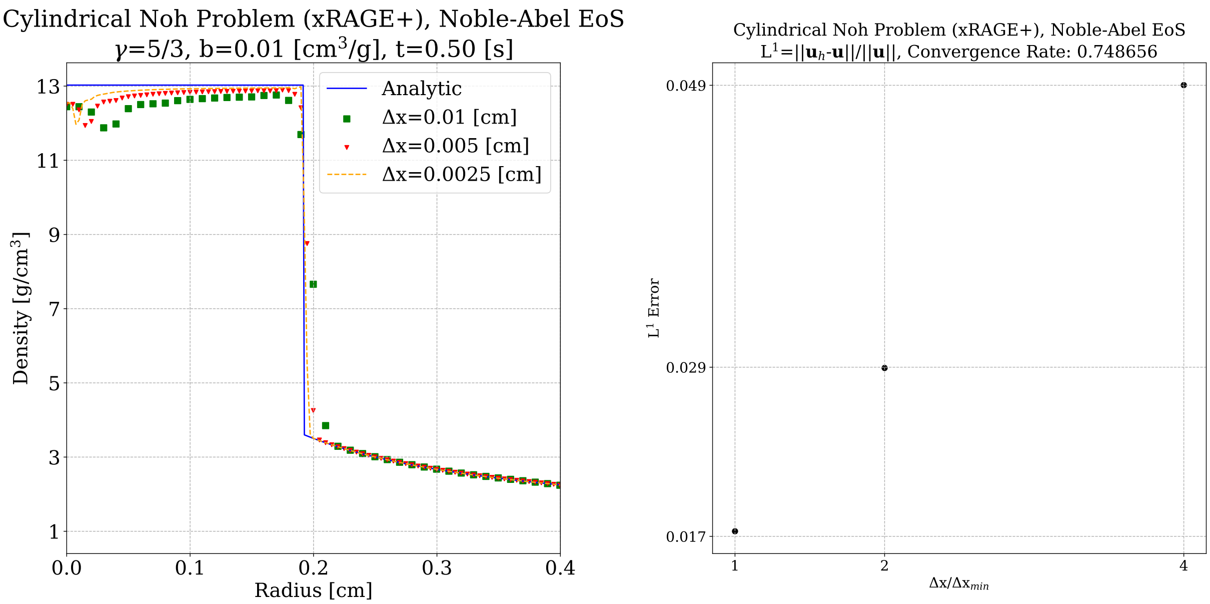

Figure 4: Noble-Abel results from xRAGE with the Singularity library. Analytic solution provided for visual comparison.

with good visual agreement to the analytic solution. Norm analysis revealed a convergence rate of one, which is consistent with non-smooth (shock containing) solutions.

Coupling of the Noble-Abel EoS in Singularity to xRAGE was successful, and results from the cylindrical Noh problem can be seen in Figure 4. Convergence rate analysis is consistent with that of Gittings et al. [2008] where it is noted that problems with shocks converge at a rate $\approx 3/4$ in xRAGE. The figure presented is a zoom fit of the simulation run, it should be noted that interactions near the boundaries yield similar results to the ideal gas case near the edge of the domain for the inflow conditions, with potential consequences propagating upstream.

5 Conclusions and Future Work

This work presented an approach to hydrodynamic code verification through the use of the Noh problem. Of particular interest was the question of how to verify correct implementation of an equation of state. Additionally, this work made contributions in the form of software that can be applied both for hydro code verification and usage. In conclusion, this work can be summarized as:

- A Python library capable of producing analytic solutions to the Noh problem and conducting error analysis.
- New equations of state implemented in the open-source library `singularity-eos`.
- Preliminary work done to scope the feasibility of obtaining solutions to the Noh problem using arbitrary equations of state.

Finding analytic solutions to the Noh problem is possible under mild assumptions on the initial conditions and the form of the equation of state. Indeed, we were able to prove that, under these assumptions, a solution exists that can easily be found via a Newton solver. These problems, broadly speaking, cover equations of state from ideal gas to Noble Abel, where the equation of state takes the form

$$e(\rho, P) = \frac{P}{\rho}\beta(\rho)$$

where β is a “nice” function. We took these theoretical results and implemented them into a python script which includes a solution library and an analysis tool for plotting and error analysis.

Our work includes some initial research regarding arbitrary or “black box” equations of state. However, future research should focus on this question: since a good deal of “workhorse” equations of state come in tabular form, being able to derive an analytical solution is of great importance. The Noh problem is a great test problem for these questions: the solution only requires us to solve the left density and pressure values, as the rest is constant and in terms of pressure and density. Naturally, finding other hydro problems with this property would provide more verification opportunities for black box EoSs that aren’t prohibitively expensive. Once that analysis is well understood, that method can be implemented into a program that can be used to run verification and validation on arbitrary equations of state.

Another avenue to pursue is when the initial pressure is non zero and potentially negative. The analysis above heavily relies on this assumption, and preliminary investigation reveals that nonzero initial pressure makes the problem much more difficult. Practically speaking, nonzero pressure is going to imply that an analytic solution will only exist in one dimension; still, it is worthwhile knowing how to construct solutions in that setting.

Acknowledgements

This work was funded by the Eulerian Applications Project (EAP) and the Cross Cutting Capabilities Project (XCAP) under the Advanced Simulation and Computing (ASC) Program at LANL. Additionally, this work would not have been possible without the guidance obtained from the project mentors.

The software (FLASH) used in this work was developed in part by the DOE NNSA and DOE Office of Science-supported Flash Center for Computational Science at the University of Chicago and the University of Rochester.

Seth J. Gerberding *is a fourth year mathematics Ph.D. student at Texas A&M university studying numerical analysis. He enjoys chess, reading, and telling engineers they need to learn more linear algebra. This has not made him many friends.*

Benjamin J. Musick *received his B.S. in Mechanical Engineering from the University of Missouri (Columbia) and is currently finishing a M.S. in Mechanical Engineering at Texas A&M University.*

Appendix A: Preliminary Thermodynamic Analysis

In an effort to continue the discussion from Section 2.5, further analysis of the constraints an EoS may place on the iterative solution approach is required. Starting from $de = TdS - Pdv$ and utilization of the well known Maxwell relations[4], we can focus on the specific derivative quantities that may challenge the solution path.

$$\begin{aligned}
\left(\frac{\partial e}{\partial P}\right)_v &= T\left(\frac{\partial S}{\partial P}\right)_v - P\left(\frac{\partial v}{\partial P}\right)_v \\
\left(\frac{\partial e}{\partial P}\right)_v &= -T\left(\frac{\partial v}{\partial T}\right)_S \\
\left(\frac{\partial e}{\partial P}\right)_v &= T\left(\frac{\partial S}{\partial T}\right)_v\left(\frac{\partial v}{\partial S}\right)_T = T\left(\frac{\partial S}{\partial T}\right)_v\left(\frac{\partial T}{\partial P}\right)_v \\
\left(\frac{\partial e}{\partial P}\right)_v &= \frac{C_v \kappa}{\alpha}
\end{aligned} \tag{41}$$

where $\alpha = \frac{1}{v}\left(\frac{\partial v}{\partial T}\right)_P$ and $\kappa = \frac{-1}{v}\left(\frac{\partial v}{\partial P}\right)_T$.

$$\begin{aligned}
\left(\frac{\partial e}{\partial v}\right)_P &= T\left(\frac{\partial S}{\partial v}\right)_P - P\left(\frac{\partial v}{\partial v}\right)_P \\
\left(\frac{\partial e}{\partial v}\right)_P &= T\left(\frac{\partial P}{\partial T}\right)_S - P \\
\left(\frac{\partial e}{\partial v}\right)_P &= -T\left(\frac{\partial P}{\partial S}\right)_T\left(\frac{\partial S}{\partial T}\right)_P - P = T\left(\frac{\partial T}{\partial V}\right)_P\left(\frac{\partial S}{\partial T}\right)_P - P \\
\left(\frac{\partial e}{\partial v}\right)_P &= \frac{C_p}{\alpha v} - P \\
\left(\frac{\partial e}{\partial \rho}\right)_P &= \frac{-1}{\rho^2}\left[\frac{C_p \rho}{\alpha} - P\right] = \frac{P}{\rho^2} - \frac{C_p}{\alpha \rho}
\end{aligned} \tag{42}$$

The two inequalities that follow the previous discussion would be,

$$\begin{aligned}
0 < \left(\frac{\partial e}{\partial P}\right)_\rho &= \frac{C_v \kappa}{\alpha} \\
0 > \left(\frac{\partial e}{\partial \rho}\right)_P &= \frac{P}{\rho^2} - \frac{C_p}{\alpha \rho}
\end{aligned} \tag{43}$$

or rearranging,

$$\begin{aligned}
0 &< \frac{C_v \kappa}{\alpha} \\
\frac{C_p \rho}{\alpha} &> P.
\end{aligned} \tag{44}$$

Future analysis will examine these properties further.

[4] https://en.wikipedia.org/wiki/Maxwell_relations

Appendix B: Extended EoS Details

The following formulations were implemented into `singularity-eos` to meet the required outputs given different combinations of input.

Stiffened Gas

Temperature

$$T(\rho, e) = \frac{\rho(e-q) - P_\infty}{\rho C_v} \tag{45}$$

Density

$$\rho(P, T) = \frac{P + P_\infty}{(\gamma - 1) C_v T} \tag{46}$$

Specific Internal Energy

$$\begin{aligned} e(\rho, T) &= \frac{\rho C_v T + P_\infty}{\rho} + q \\ e(P, T) &= \frac{P + \gamma P_\infty}{P + P_\infty} C_v T + q \\ e(P, \rho) &= \frac{P + \gamma P_\infty}{\rho(\gamma - 1)} + q \end{aligned} \tag{47}$$

Pressure

$$\begin{aligned} P(\rho, T) &= \rho(\gamma - 1) C_v T - P_\infty \\ P(\rho, e) &= \rho(e - q)(\gamma - 1) - \gamma P_\infty \end{aligned} \tag{48}$$

Entropy Availability

$$\begin{aligned} \Delta S(\rho, T) &= C_v \ln\left(\frac{T}{T_0}\right) + (\gamma - 1) C_v \ln\left(\frac{\rho_0}{\rho}\right) + q' \\ \Delta S(\rho, e) &= C_v \ln\left(\frac{e - q - P_\infty v}{e_0 - q - P_\infty v_0}\right) + (\gamma - 1) C_v \ln\left(\frac{\rho_0}{\rho}\right) + q' \end{aligned} \tag{49}$$

Specific Heat at Constant Volume $\left(\left(\frac{\partial e}{\partial T}\right)_v\right)$

$$\begin{aligned} C_v(\rho, T) =& C_v \\ C_v(\rho, e) =& C_v \end{aligned} \tag{50}$$

Isentropic Bulk Modulus $\left(\rho\left(\frac{\partial P}{\partial \rho}\right)_S\right)$

$$\begin{aligned} \rho\left(\frac{\partial P}{\partial \rho}\right)_S &= \rho(e - q)(\gamma - 1) + P\Gamma \\ B_s(\rho, T) &= \Gamma \gamma \rho C_v T \\ B_s(\rho, e) &= \Gamma \gamma \left[\rho(e - q) - P_\infty\right] \end{aligned} \tag{51}$$

Gruneisen Parameter $\left(\frac{1}{\rho}\left(\frac{\partial P}{\partial e}\right)_\rho\right)$

$$\begin{aligned} \Gamma(\rho, T) =& \gamma - 1 \\ \Gamma(\rho, e) =& \gamma - 1 \end{aligned} \tag{52}$$

Noble-Abel

Temperature

$$T(\rho, e) = \frac{e - q}{C_v} \tag{53}$$

Density

$$\rho(P,T) = \frac{P}{C_v(\gamma-1)T + bP} \tag{54}$$

Specific Internal Energy

$$\begin{aligned} e(\rho,T) &= C_vT + q \\ e(P,T) &= C_vT + q \\ e(P,\rho) &= \frac{P(1-b\rho)}{\gamma-1} + q \end{aligned} \tag{55}$$

Pressure

$$\begin{aligned} P(\rho,T) &= \frac{\rho C_v T(\gamma-1)}{1-b\rho} \\ P(\rho,e) &= \frac{\rho(e-q)(\gamma-1)}{1-b\rho} \end{aligned} \tag{56}$$

Entropy Availability

$$\begin{aligned} \Delta S(\rho,T) &= C_v \ln\left(\frac{T}{T_0}\right) + C_v(\gamma-1)\ln\left(\frac{v-b}{v_0-b}\right) + q' \\ \Delta S(\rho,e) &= C_v \ln\left(\frac{e-q}{e_0-q}\right) + C_v(\gamma-1)\ln\left(\frac{v-b}{v_0-b}\right) + q' \end{aligned} \tag{57}$$

Specific Heat at Constant Volume $\left(\left(\frac{\partial e}{\partial T}\right)_v\right)$

$$\begin{aligned} C_v(\rho,T) &= C_v \\ C_v(\rho,e) &= C_v \end{aligned} \tag{58}$$

Isentropic Bulk Modulus $\left(\left(\frac{\partial P}{\partial \rho}\right)_S\right)$

$$\begin{aligned} \rho\left(\frac{\partial P}{\partial \rho}\right)_S &= \rho\frac{(e-q)(\gamma-1)}{(1-b\rho)^2} + P\Gamma \\ B_s(\rho,T) &= \Gamma\frac{\rho C_v T\gamma}{(1-b\rho)} \\ B_s(\rho,e) &= \Gamma\frac{\rho(e-q)\gamma}{(1-b\rho)} \end{aligned} \tag{59}$$

Gruneisen Parameter $\left(\frac{1}{\rho}\left(\frac{\partial P}{\partial e}\right)_\rho\right)$

$$\begin{aligned} \Gamma(\rho,T) &= \frac{\gamma-1}{1-b\rho} \\ \Gamma(\rho,e) &= \frac{\gamma-1}{1-b\rho} \end{aligned} \tag{60}$$

Carnahan-Starling

Compressibility Factor

$$\begin{aligned} Z(\rho) &= \frac{1+\eta+\eta^2-\eta^3}{(1-\eta)^3} \\ \eta &= b\rho \\ \rho Z(\rho) &= \frac{\rho + b\rho^2 + b^2\rho^3 - b^3\rho^4}{(1-b\rho)^3} \\ \frac{Z(v)}{v} &= \frac{v^3 + bv^2 + b^2v - b^3}{v(v-b)^3} = \frac{1}{v} + \frac{4b}{(v-b)^2} + \frac{2b^2}{(v-b)^3} \\ \frac{\partial(\rho Z)}{\partial \rho} &= \frac{b^4\rho^4 - 4b^3\rho^3 + 4b^2\rho^2 + 4b\rho + 1}{(1-b\rho)^4} \end{aligned} \tag{61}$$

Temperature

$$T(\rho, e) = \frac{e - q}{C_v} \tag{62}$$

Density (Newton's Method or `RootFinding1D::findRoot` in `singularity-eos`)

$$\begin{aligned}
f(\rho) = 0 &= (\rho Z) T C_v(\gamma - 1) - P \\
f'(\rho) &= \frac{\partial(\rho Z)}{\partial \rho} T C_v(\gamma - 1) \\
\rho^{n+1}(P, T) &= \rho^n - f(\rho^n) / \left[\frac{\partial f(\rho^n)}{\partial \rho}\right]
\end{aligned} \tag{63}$$

Specific Internal Energy

$$\begin{aligned}
e(\rho, T) &= C_v T + q \\
e(P, T) &= C_v T + q \\
e(P, \rho) &= \frac{P}{Z\rho(\gamma - 1)} + q
\end{aligned} \tag{64}$$

Pressure

$$\begin{aligned}
P(\rho, T) &= Z\rho T C_v(\gamma - 1) \\
P(\rho, e) &= Z\rho(e - q)(\gamma - 1)
\end{aligned} \tag{65}$$

Entropy Availability

$$\begin{aligned}
\Delta S(\rho, T) = {} & C_v \ln\left(\frac{T}{T_0}\right) + C_v(\gamma - 1) \ln\left(\frac{\rho_0}{\rho}\right) \\
& - C_v(\gamma - 1)\left(\frac{4b}{v - b} - \frac{4b}{v_0 - b} + \frac{b^2}{(v - b)^2} - \frac{b^2}{(v_0 - b)^2}\right) + q'
\end{aligned} \tag{66}$$

$$\begin{aligned}
\Delta S(\rho, e) = {} & C_v \ln\left(\frac{e - q}{e_0 - q}\right) + C_v(\gamma - 1) \ln\left(\frac{\rho_0}{\rho}\right) \\
& - C_v(\gamma - 1)\left(\frac{4b}{v - b} - \frac{4b}{v_0 - b} + \frac{b^2}{(v - b)^2} - \frac{b^2}{(v_0 - b)^2}\right) + q'
\end{aligned} \tag{67}$$

Specific Heat at Constant Volume $\left(\left(\frac{\partial e}{\partial T}\right)_v\right)$

$$\begin{aligned}
C_v(\rho, T) = {} & C_v \\
C_v(\rho, e) = {} & C_v
\end{aligned} \tag{68}$$

Isentropic Bulk Modulus $\left(\left(\frac{\partial P}{\partial \rho}\right)_S\right)$

$$\begin{aligned}
\rho\left(\frac{\partial P}{\partial \rho}\right)_S &= \rho(e - q)(\gamma - 1)\frac{\partial(\rho Z)}{\partial \rho} + P\Gamma \\
B_s(\rho, T) &= \rho C_v T(\gamma - 1)\left[\frac{\partial(\rho Z)}{\partial \rho} + Z\Gamma\right] \\
B_s(\rho, e) &= \rho(e - q)(\gamma - 1)\left[\frac{\partial(\rho Z)}{\partial \rho} + Z\Gamma\right]
\end{aligned} \tag{69}$$

Gruneisen Parameter $\left(\Gamma = \frac{1}{\rho}\left(\frac{\partial P}{\partial e}\right)_\rho\right)$

$$\begin{aligned}
\Gamma(\rho, T) &= Z(\gamma - 1) \\
\Gamma(\rho, e) &= Z(\gamma - 1)
\end{aligned} \tag{70}$$

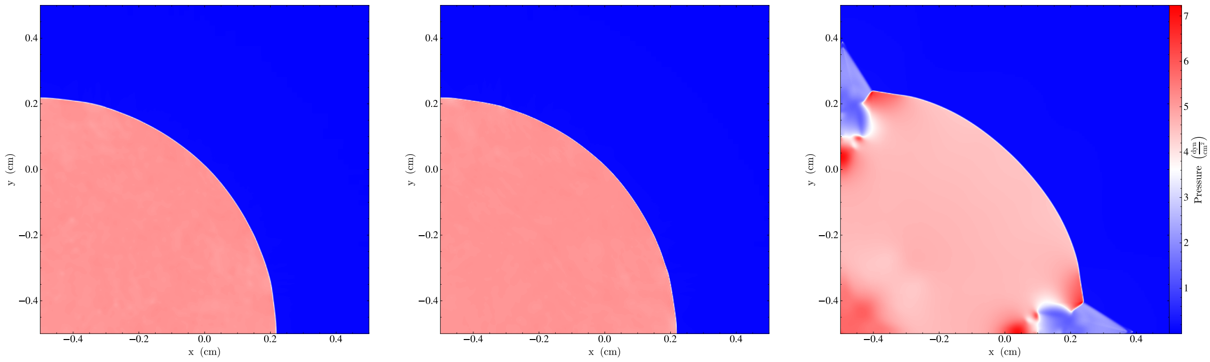

Figure 5: An example of a 2D, cylindrical Noh problem run for a monatomic, calorically perfect gas in the FLASH4.7 hydrodynamics code Fryxell et al. [2000]. Note the analytic solution predicts a constant post-shock pressure of $5.\bar{3}$. (Left) Split, 3^{rd} order PPM flux reconstruction, Colella and Glaz type Riemann solver, CFL=0.8. (Middle) Unsplit, 5^{th} order WENO flux reconstruction, van Leer slope limiter, HLLC Riemann sovler, CFL=0.8. (Right) Unsplit 1^{st} order Godunov flux reconstruction, van Leer slope limiter, Roe Riemann solver, CFL=0.2.

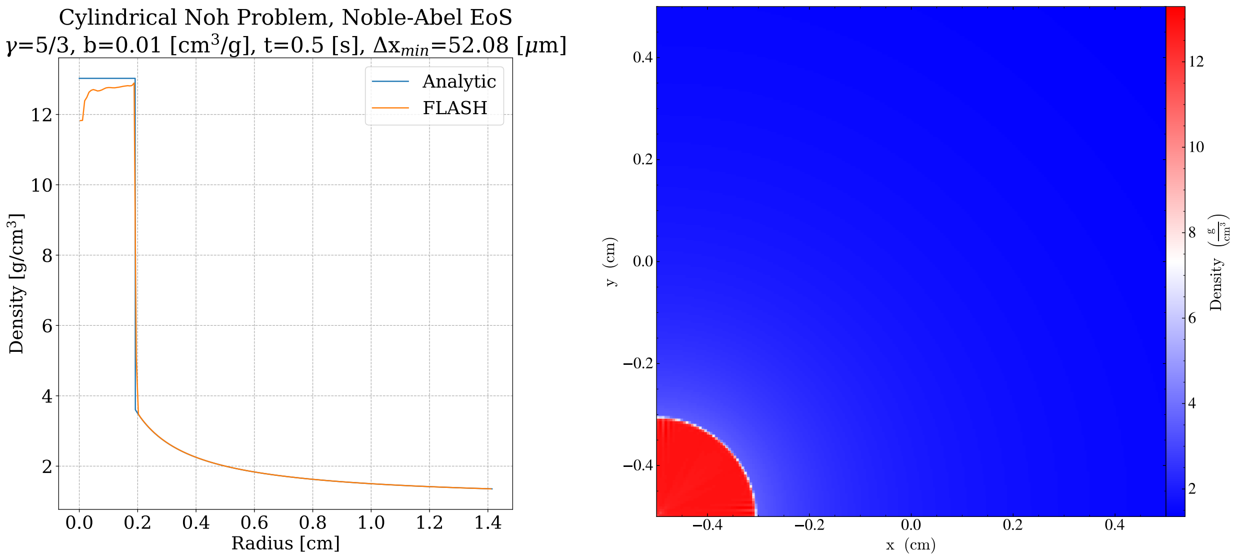

Figure 6: Cylindrical Noh: Noble-Abel EoS

Appendix C: Noh Simulations (FLASH)

The FLASH4.7 hydro code Fryxell et al. [2000] was utilized as a testing ground to test the validity of the iterative solution process for planar Noh cases. In addition, the cylindrical Noh problem using a Noble-Abel EoS was tested to utilize the analytic solution library. Results can be seen in Figures 6 through 12.

FLASH simulations were run with unsplit hydrodynamics, 5^{th} order WENO flux reconstruction, HLLC Riemann solver, and a CFL=0.2. Uniform initial conditions of $\rho_R = 1$, $u_R = -1$, and $P_R = 0$. EoS parameters and solution found in Table 2. Uniform mesh chosen, coarse resolution of $1/\Delta x$=48 and fine resolution of $1/\Delta x$=768.

Equations of state not implemented in FLASH were done so through modification of the calorically perfect ideal gas, single gamma EoS (`Gamma`). Sound speed calculations were modified in the HLLC Riemann solver and the general eigenvalue routine for computing wave speeds when applicable (see files *hy_uhd_HLLC.F90* and *hy_uhd_eigenParameters.F90*). Use of the SESAME Tables was done through the Multi-Type, Multi-Material, Multi-Temperature (`+mtmmmt`) closure model using the `Tabulated` EoS.

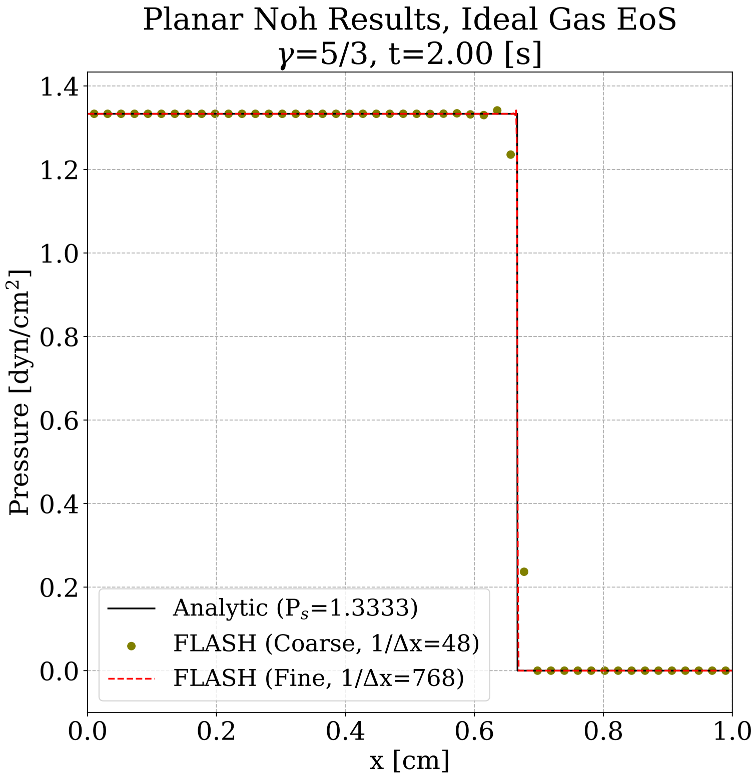

Figure 7: Planar Noh: Ideal Gas EoS.

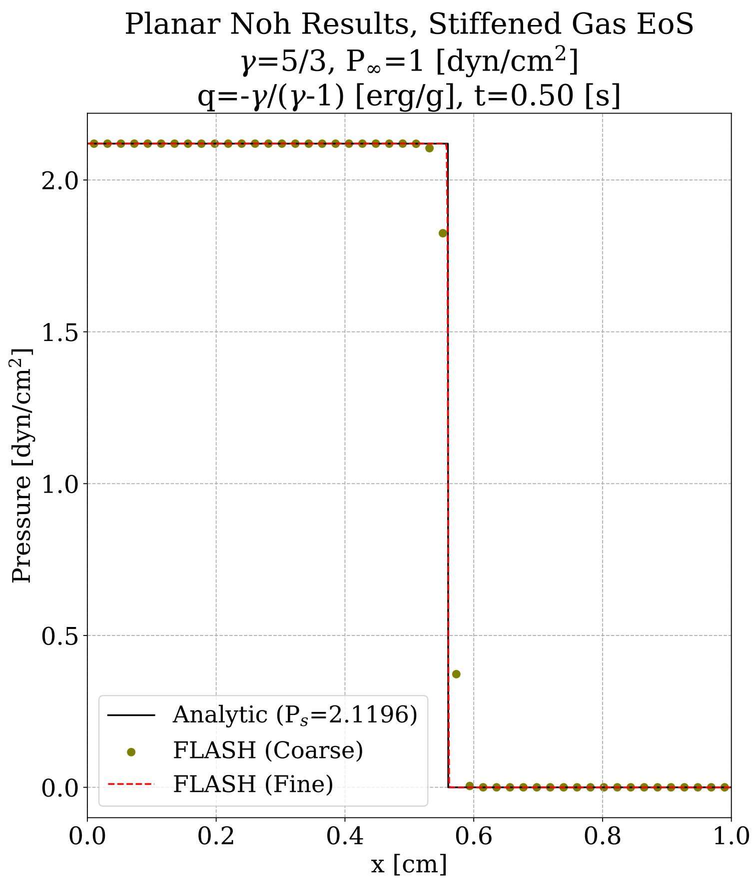

Figure 8: Planar Noh: Stiffened Gas EoS.

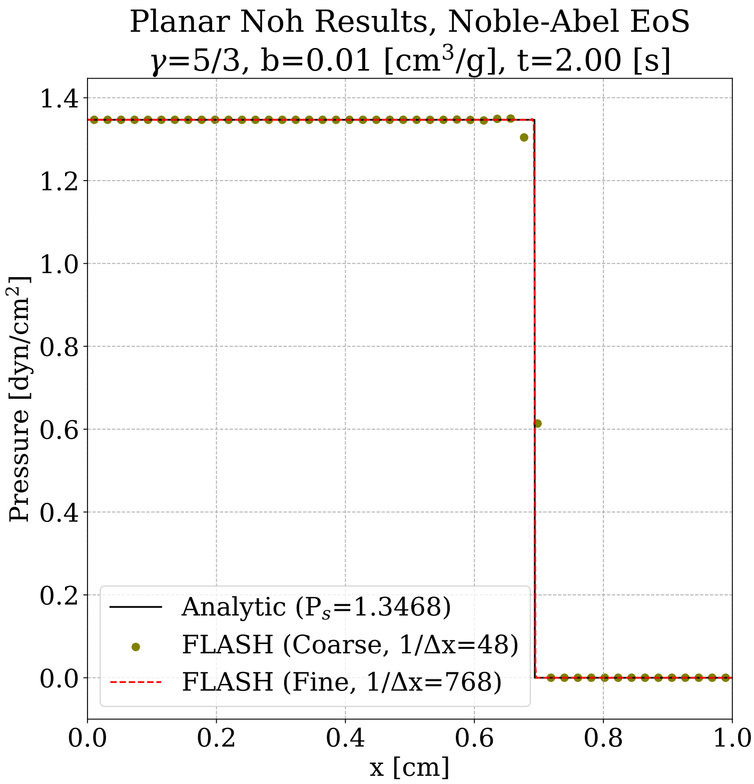

Figure 9: Planar Noh: Noble-Abel EoS.

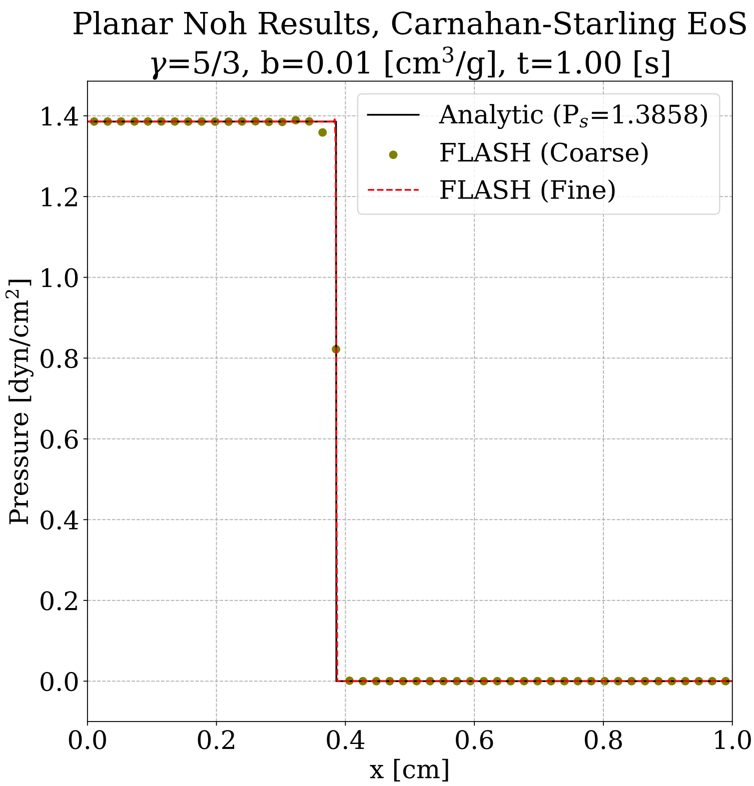

Figure 10: Planar Noh: Carnahan-Starling EoS.

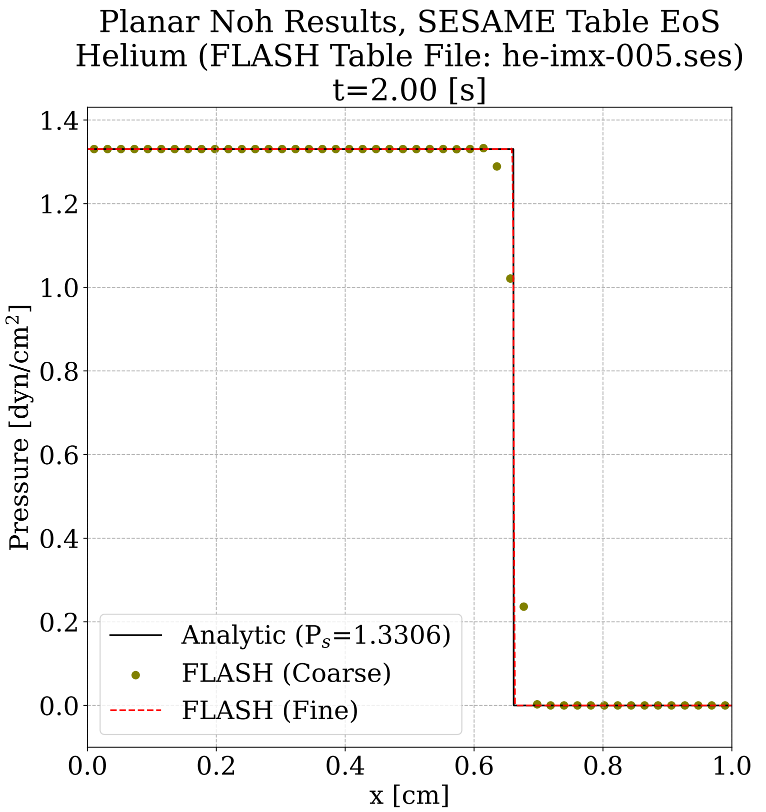

Figure 11: Planar Noh: SESAME Table EoS, He.

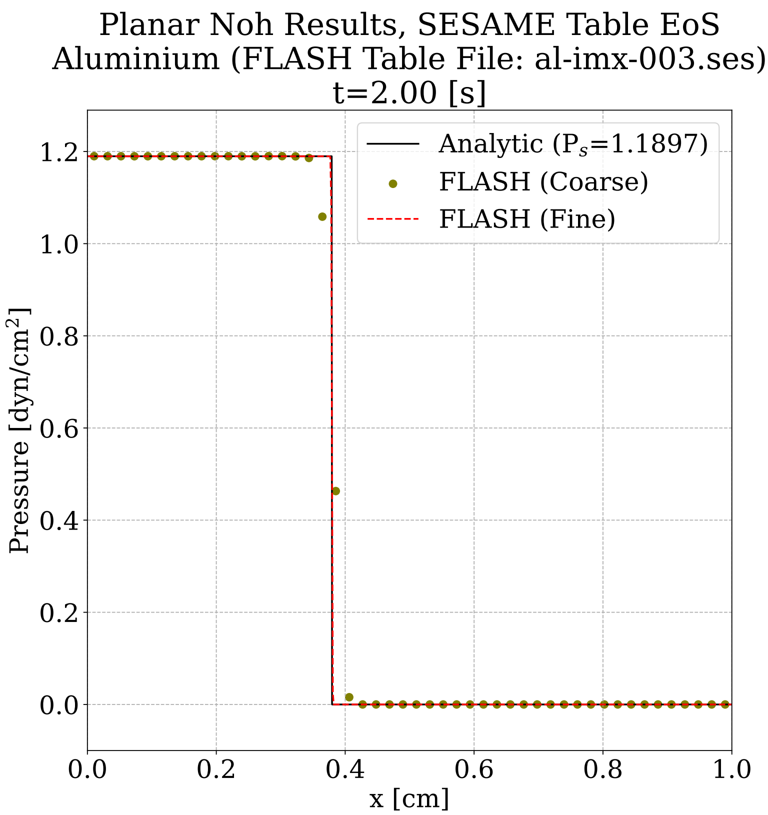

Figure 12: Planar Noh: SESAME Table EoS, Al.

References

Roy A. Axford. Solutions of the noh problem for various equations of state using lie groups. *Laser and Particle Beams*, 18(1):93–100, 2000. doi: 10.1017/S026303460018111X.

Sarah C. Burnett, Kevin G. Honnell, Scott D. Ramsey, and Robert L. Singleton Jr. Verification studies for the noh problem using non-ideal equations of state and finite strength shocks. *Journal of Verification, Validation and Uncertainty Quantification*, 2017.

W. P. Crowley. Flag: A free-lagrange method for numerically simulating hydrodynamic flows in two dimensions. In Maurice Holt, editor, *Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics*, pages 37–43, Berlin, Heidelberg, 1971. Springer Berlin Heidelberg. ISBN 978-3-540-36533-4.

B. Fryxell, K. Olson, P. Ricker, F. X. Timmes, M. Zingale, D. Q. Lamb, P. MacNeice, R. Rosner, J. W. Truran, and H. Tufo. Flash: An adaptive mesh hydrodynamics code for modeling astrophysical thermonuclear flashes. *The Astrophysical Journal Supplement Series*, 131(1):273, nov 2000. doi: 10.1086/317361. URL `https://dx.doi.org/10.1086/317361`.

Michael Gittings, Robert Weaver, Michael Clover, Thomas Betlach, Nelson Byrne, Robert Coker, Edward Dendy, Robert Hueckstaedt, Kim New, W Rob Oakes, Dale Ranta, and Ryan Stefan. The rage radiation-hydrodynamic code. *Computational Science & Discovery*, 1(1):015005, nov 2008. doi: 10.1088/1749-4699/1/1/015005. URL `https://dx.doi.org/10.1088/1749-4699/1/1/015005`.

O Le Métayer, J Massoni, and R Saurel. Élaboration des lois d'état d'un liquide et de sa vapeur pour les modèles d'écoulements diphasiques. *International Journal of Thermal Sciences*, 43(3):265–276, 2004. ISSN 1290-0729. doi: https://doi.org/10.1016/j.ijthermalsci.2003.09.002. URL `https://www.sciencedirect.com/science/article/pii/S1290072903001443`.

Richard Liska and Burton Wendroff. Comparison of several difference schemes on 1d and 1d test problems for the euler equations. *SIAM Journal of Scientific Computation*, 24(3):995–1107, 2003.

Loann Neron and Richard Saurel. Noble–Abel/first-order virial equations of state for gas mixtures resulting from multiple condensed reactive materials combustion. *Physics of Fluids*, 34(1):016107, 01 2022. ISSN 1070-6631. doi: 10.1063/5.0079187. URL `https://doi.org/10.1063/5.0079187`.

W.F Noh. Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux. *Journal of Computational Physics*, 72(1):78–120, 1987. ISSN 0021-9991. doi: https://doi.org/10.1016/0021-9991(87)90074-X. URL `https://www.sciencedirect.com/science/article/pii/002199918790074X`.

Jeffrey Hammett Peterson, Jonah Maxwell Miller, Anna Mataleena Pietarila Graham, Daniel Alphin Holladay, Christopher Michael Mauney, Richard Felix Berger, and Karen Chung-Yen Tsai. Singularity-eos xcap report. *Los Alamos Natl. Lab.*, 12 2022. doi: 10.2172/1907763. URL `https://www.osti.gov/biblio/1907763`.

S. D. Ramsey, Z. M. Boyd, and S. C. Burnett. Solution of the noh problem using the universal symmetry of the gas dynamics equations. *Shock Waves*, 27(3):477–485, 2017. doi: 10.1007/s00193-016-0670-z. URL `https://doi.org/10.1007/s00193-016-0670-z`.

Scott D. Ramsey, Emma M. Schmidt, Zachary M. Boyd, Jennifer F. Lilieholm, and Roy S. Baty. Converging shock flows for a Mie-Grüneisen equation of state. *Physics of Fluids*, 30(4):046101, 04 2018. ISSN 1070-6631. doi: 10.1063/1.5018323. URL `https://doi.org/10.1063/1.5018323`.

Chloe' Yorke, April Howard, Sarah Burnett, Kevin Honnell, Scott Ramsey, and Robert Singleton. Extension of the planar noh problem to aluminum, iron, copper, and tungsten. 2018.