Anatomy of a Multi-Code Verification Test System

Scott W. Doebling, Ph.D.
Group Leader, Verification & Analysis Group (XCP-8)
Computational Physics Division

Collaborators: Dan Israel, Jim Kamm, Bob Singleton

May 2014
A multi-code verification test system

- Perform code verification in a consistent manner
- Across multiple computational physics codes
- Overview of components and brief discussion
- Future talks (one today!) will describe the components in depth
Why a multi-code verification test system?

- Enable standardized code verification testing
- Objective evaluation of accuracy & convergence
- Objective basis for comparison across codes
- Ensure consistency and repeatability
- Multi-code verification is not “code-to-code comparison”
Anatomy of a multi-code verification test system

- Definitions for verification test problems
- Test problem exact solutions
- Input deck generation & simulation execution
- Code verification analysis
- Documentation & archiving
Definitions for verification test problems

- Specification of problem parameters
- Specification of computational setup
  - Code independent, e.g. Lagrangian vs Eulerian, to extent possible
- Specification of code outputs
Definitions for verification test problems

- Proposed standard definitions for Sedov, Noh, and Riemann published
  - Attached to the end of this document!

- Revised versions will be published as we add more problems.
Example of standardized definition for verification test problem

**Sedov Problem**

**Description:** The Sedov Problem is a mathematical idealization of a shock generated via an explosion. It consists of spherically symmetric flow of an inviscid, non-heat conducting, compressible, polytropic gas, driven by a single zone with non-trivial initial energy. This problem tests a code’s ability to convert internal energy into kinetic energy and has a quasi-analytic, self-similar solution that requires one numerical quadrature.

**Figure 1:** Initial configuration of the Sedov problem.

<table>
<thead>
<tr>
<th>$c_m$ [s]</th>
<th>$\gamma$ [-]</th>
<th>$\rho_0$ [g/cm$^3$]</th>
<th>$u_0$ [cm/s]</th>
<th>$p_0$ [dyn/cm$^2$]</th>
<th>Internal Energy [erg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>7/5</td>
<td>1.0</td>
<td>0.0</td>
<td>$2(5)/10^{-12}$</td>
<td>1D: 0.851072 2D: 0.425536</td>
</tr>
</tbody>
</table>

**Mesh:** $R_{min} = r_{min} = 0.0$, $R_{max} = r_{max} = 1.2$ cm; in 2D, $x_{min} = 0.0$, $x_{max} = 1.2$ cm.

1D spherical: $N_r = 60, 120, 240, 480$  
2D cylindrical: $N_r = N_z = 60, 120, 240, 480$

**Table 2:** SIE [erg/g] in first zone that gives internal energy in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>60</th>
<th>120</th>
<th>240</th>
<th>480</th>
</tr>
</thead>
<tbody>
<tr>
<td>1D</td>
<td>2.5397311x10$^4$</td>
<td>2.0317849x10$^5$</td>
<td>1.6254279x10$^6$</td>
<td>1.3003423x10$^7$</td>
</tr>
<tr>
<td>2D</td>
<td>1.6931541x10$^4$</td>
<td>1.3545233x10$^5$</td>
<td>1.0836186x10$^6$</td>
<td>8.6689490x10$^6$</td>
</tr>
</tbody>
</table>

**Figure 2:** Sedov Problem radial velocity.

**Figure 3:** Sedov Problem pressure.

**Figure 4:** Sedov Problem SIE.
Test problem exact solutions

- Code verification requires highly accurate – preferably “exact” - problem solutions
- For shock hydrodynamics, many individual exact solution codes have been written, but:
  - They exist in disparate locations
  - No standardized input or output format
  - Users must add own plotting package, etc.
- Need a single, standardized toolbox for these codes
**ExactPack: An Exact Solution Analysis Package**

- New software package for exact solution codes
- Enables plotting of exact solutions
- Easily extended to new problems
- Use as stand-alone code or as a python package
- Code is robust, reliable, and maintainable
- Currently contains several compressible hydro problems: Noh, Sedov, Riemann, Guderley, …
- Will be available soon for public usage. Presentation by Bob Singleton immediately following!
Input deck generation & simulation management

- Need a reliable, repeatable, robust way to generate code input decks
- Consistency across codes with arbitrarily different input formats
- Use LANL “common model framework” (CMF) templates
- Problem-specific information stored in LANL V&V System (VVS) repository
- Build documentation from this repository
Using CMF templates and VVS problem definitions for a code verification study

From VVS repository
- Problem File (code independent/problem specific)
  e.g. grid sizes and problem parameters
- Code File (code specific/problem specific)
  e.g. hydro options, BC’s

From CMF repository
- Input Templates (code specific/problem independent)
- Run specification (code specific/problem specific)
- Suite Results (code specific/problem specific)

Build Decks
- Suite Input Decks (code specific/problem specific)

Run Calculations

Analyze & Document

LA-UR-14-23068
Verification Pedigree and Reproducibility

- Pedigree of verification test suite results
  - Which problem
  - With what code settings
  - Done by who and when

- Ability to reproduce code test results:
  - Same problem in same code with specified version
  - Same problem in different code
We are developing a standardized method for code verification analysis within the ASC program at LANL.

**Workflow**

- **Perform calculations on series of grids**
- **Generate exact solution**
- **Compare physical variables**
- **Evaluate global error metrics**
- **Perform convergence analysis**
- **Present verification results**

**Choices to be made**

- Problem definitions
- Input decks
- Grids

**ExactPack**

- Metrics
- Weighting
- Mapping between grids

**Standardize**

- Error model
- Method for calculating convergence rate

- Plot format
- Report format

**Standardize** (et al.)
Working on a standardized format for presenting convergence results

- Problem name & parameters
- Code name, version & options
- User name & date

Legend contains:
- Variable name
- Error norm weight
- Convergence rate
- R-value for fit

Axis limits are integer powers of 10

Point values for error norm, with fit line (unique markers and colors)

Convergence fiducial

Log-log plot with gridlines

y-axis is error norm

x-axis is length scale (not number of zones)

Problem: Sedov, $\gamma = 5/3$
Code: FLAG 3.4.0, MARS-1, r-z grid
User: doebling@lanl.gov, 4/23/14
Automated documentation

- Sphinx tools with RST markup language
- Store problem definition and skeleton of analysis report in repository
- Generate analysis results using standardized script
- Human-readable report in HTML, PDF, LaTex, etc.
- Generate “stoplight” metrics as required
- Distribute results to stakeholders
- Archive results at some interval (e.g. time-based or release-based)
A multi-code verification test system

- Perform code verification in a consistent manner
- Across multiple computational physics codes
- Overview of components and brief discussion
- Future talks (one today!) will describe the components in depth

Thank you, you've been a great audience!
Title: Standardized Definitions for Code Verification Test Problems

Author(s): Kamm, James R.
Doebling, Scott W.
Israel, Daniel M.
Singleton, Robert

Intended for: For research collaborators
Report

Issued: 2014-05-02 (rev.1)
Standardized Definitions for Code Verification Test Problems

James R. Kamm*, Scott W. Doebling, Daniel M. Israel, Robert Singleton

Los Alamos National Laboratory, Los Alamos, NM 87545 USA

Executive Summary

This document contains standardized definitions of code verification test problems used in the quantitative evaluation of compressible flow algorithms. The problems documented are:

1. The Finite Noh Problem in spherical geometry
2. The Sedov Problem in spherical geometry
3. Six 1D Riemann Problem in planar geometry

The definitions provided are intended to enable users to set up and simulate each problem in a computational physics code in a consistent manner. The exact solution results shown in the figures enable the user to perform a qualitative visual confirmation of the accuracy of the computed results.

* Corresponding author. Email: jrkamm@lanl.gov
**Finite Noh Problem**

*Description:* The Finite Noh Problem is a finite-domain restriction of the mathematically ideal, infinite domain, spherically symmetric, infinite-strength shock impinging on a rigid wall. It consists of an inviscid, non-heat conducting, compressible, polytropic gas, initialized with a uniform, spherically radially inward velocity, \( u_{R,0} \). This problem tests a code’s ability to convert kinetic energy into internal energy.

![Initial state: \( \rho_0, u_{R,0}, p_0 \)](image)

![Initial configuration of the spherical Noh problem.](image)

| 1D spherical symmetric | 2D cylindrically symmetric |

<table>
<thead>
<tr>
<th>r</th>
<th>z_max</th>
<th>z_min</th>
</tr>
</thead>
<tbody>
<tr>
<td>r</td>
<td>Cylindrically symmetric</td>
<td>Planar symmetric</td>
</tr>
<tr>
<td>r</td>
<td>Initial state: ( \rho_0, u_{R,0}, p_0 )</td>
<td>Initial state: ( \rho_0, u_{R,0}, p_0 )</td>
</tr>
</tbody>
</table>

Table 1: Parameters for the Noh problem.

<table>
<thead>
<tr>
<th></th>
<th>( t_{in} ) [s]</th>
<th>( \gamma ) [-]</th>
<th>( \rho_0 ) [g/cm(^3)]</th>
<th>( u_0 ) [cm/s]</th>
<th>( p_0 ) [dyn/cm(^2)]</th>
<th>( E_0 ) [erg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1D</td>
<td>0.6</td>
<td>5/3</td>
<td>1.0</td>
<td>( u_{R,0} = -1.0 )</td>
<td>( (2/3) \times 10^{-12} )</td>
<td>( 1 \times 10^{-12} )</td>
</tr>
<tr>
<td>2D</td>
<td>0.6</td>
<td>5/3</td>
<td>1.0</td>
<td>( u_{r,0} = -r/R, \ u_{z,0} = -z/R )</td>
<td>( (2/3) \times 10^{-12} )</td>
<td>( 1 \times 10^{-12} )</td>
</tr>
</tbody>
</table>

**Mesh\(^*\):** \( R_{min} = r_{min} = 0.0, R_{max} = r_{max} = 1.2 \) cm; in 2D, \( z_{min} = 0.0, z_{max} = 1.2 \) cm.

1D spherical: \( N_R = 60, 120, 240, 480 \)  
2D cylindrical: \( N_r = N_z = 60, 120, 240, 480 \)  
1D: \( \Delta R = 0.02, 0.01, 0.005, 0.0025 \)  
2D: \( \Delta r = \Delta z = 0.02, 0.01, 0.005, 0.0025 \)

**Initial conditions:** Uniform and constant material density, pressure, and spherically radial velocity. States are related through the polytropic (ideal gas) EOS:

\[ p = (\gamma - 1) \rho e \]

where \( \gamma = 5/3 \) is the (constant) ratio of specific heats.

**Boundary conditions:** The problem is to be run so that any spurious waves generated by boundary conditions do not affect the solution on the Comparison Domain, defined below. Possible boundary conditions are:

Inner: \( R_{min}, r_{min}, z_{min} \): Symmetric  
Outer: \( R_{max}, r_{max}, z_{max} \): Constant in time

Table 2: Problem solution at \( t_{fin}=0.6 \) s:

<table>
<thead>
<tr>
<th>r [cm]</th>
<th>( \rho ) [g/cm(^3)]</th>
<th>( u_r ) [cm/s]</th>
<th>( p ) [dyn/cm(^2)]</th>
<th>( e ) [erg/g]</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R &lt; R_5 = 0.2 )</td>
<td>64.0</td>
<td>0.0</td>
<td>( 21_3^1 )</td>
<td>1/2</td>
</tr>
<tr>
<td>( R &gt; R_5 = 0.2 )</td>
<td>([1+(0.6/R)]^2 )</td>
<td>-1.0</td>
<td>( (2/3) \rho \times 10^{-12} )</td>
<td>( 1 \times 10^{-12} )</td>
</tr>
</tbody>
</table>

**Output**, in ASCII comma- or space-delimited format\(^\dagger\), to include:

i. Values of density, velocity, pressure, SIE as a function of position at \( t_0 \) and \( t_{fin} \).

ii. Entire mesh total energy, kinetic energy, internal energy as a function of time.

**Comparison Domain:**  
1D: \( R \in [0, 0.5] \)  
2D: \( (r, z) \in [0, 0.5] \times [0, 0.5] \)

\(^*\) Here, \( R \) = spherical radial coordinate, with \( R^2 = r^2 + z^2 \), where \( r \) = cylindrical radial coordinate.

\(^\dagger\) Sample output available upon request.
Finite Noh Problem Results at $t_{\text{fin}} = 0.6$ s

Figure 1. Noh Problem density.

Figure 2. Noh Problem radial velocity.

Figure 3. Noh Problem pressure.

Figure 4. Noh Problem SIE.
Sedov Problem

Description: The Sedov Problem is an idealization of a blast wave generated via an explosion. It consists of spherically symmetric flow of an inviscid, non-heat conducting, compressible, polytropic gas, initially driven by a single zone with non-trivial internal energy. This problem tests a code’s ability to convert internal energy into kinetic energy and has a quasi-analytic, self-similar solution.

Mesh*: \( R_{\min} = r_{\min} = 0.0, \ R_{\max} = r_{\max} = 1.2 \) cm; in 2D, \( z_{\min} = 0.0, z_{\max} = 1.2 \) cm.

1D spherical: \( N_r = 60, 120, 240, 480 \) 2D cylindrical: \( N_r = N_z = 60, 120, 240, 480 \)

1D: \( \Delta R = 0.02, 0.01, 0.005, 0.0025 \) 2D: \( \Delta r = \Delta z = 0.02, 0.01, 0.005, 0.0025 \)

Table 1: Parameters for the Sedov problem.

<table>
<thead>
<tr>
<th>( t_{\text{fin}} ) [s]</th>
<th>( \gamma ) [-]</th>
<th>( \rho_0 ) [g/cm(^3)]</th>
<th>( u_0 ) [cm/s]</th>
<th>( \rho_0 ) [dyn/cm(^2)]</th>
<th>Internal Energy [erg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>7/5</td>
<td>1.0</td>
<td>0.0</td>
<td>(2/5) \times 10^{-12}</td>
<td>1D: 0.851072 2D: 0.425536</td>
</tr>
</tbody>
</table>

Table 2: SIE [erg/g] in first zone that gives internal energy in Table 1.

<table>
<thead>
<tr>
<th>( R )</th>
<th>( \rho_0 )</th>
<th>( u_0 )</th>
<th>( \rho_0 )</th>
<th>Internal Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>2.5397311 \times 10^4</td>
<td>2.0317849 \times 10^5</td>
<td>1.6254279 \times 10^6</td>
<td>1.3003423 \times 10^7</td>
</tr>
<tr>
<td>120</td>
<td>1.6931541 \times 10^4</td>
<td>1.3545233 \times 10^5</td>
<td>1.0836186 \times 10^6</td>
<td>8.6689490 \times 10^6</td>
</tr>
</tbody>
</table>

Initial conditions: Uniform and constant material density and velocity. SIE in the single energetic zone per Table 2, elsewhere \( 10^{-12} \) erg/g, so that pressure and total energy equal values in Table 1. States are related through the polytropic (ideal gas) EOS: \( p = (\gamma - 1)\rho e \) where \( \gamma = 7/5 \) is the (constant) ratio of specific heats.

Boundary conditions: The problem is to be run so that any spurious waves generated by boundary conditions do not affect the solution on the Comparison Domain, defined below. Possible boundary conditions are:

Inner: \( R_{\min}, r_{\min}, z_{\min} \): Symmetric  
Outer: \( R_{\max}, r_{\max}, z_{\max} \): Constant in time

Output, in ASCII comma- or space-delimited format†, to include:

i. Values of density, velocity, pressure, SIE as a function of position at \( t_0 \) and \( t_{\text{fin}} \).

ii. Entire mesh total energy, kinetic energy, internal energy as a function of time.

Comparison Domain: 1D: \( R \in [0,1.2] \) 2D: \( (r,z) \in [0,1.2] \times [0,1.2] \)

* Here, \( R \) = spherical radial coordinate, with \( R^2 = r^2 + z^2 \), where \( r \) = cylindrical radial coordinate.
† Sample output available upon request.
Sedov Problem Results at $t_{\text{fin}} = 1$ s

Figure 1: Sedov Problem density.

Figure 2: Sedov Problem radial velocity.

Figure 3. Sedov Problem pressure.

Figure 4. Sedov Problem SIE.
1D Riemann Problems

Description: 1D Riemann Problems are finite-domain restrictions of mathematically ideal, infinite domain shock tube experiments. They are in Cartesian geometry with two materials separated by a massless interface. At \( t=0 \), the states are constant and uniform and uniform. The removal of the interface leads to the evolution of the self-similar solution, consisting of some combination of shock, contact, and rarefaction waves.

\[
\begin{array}{|c|c|c|}
\hline
y_{\text{max}} & y_{\text{min}} & x_{\text{min}} \\
\hline
 & & x_{\text{int}} \\
\hline
 & & x_{\text{max}} \\
\hline
\end{array}
\]

Initial geometry of Cartesian Riemann problems.

The tests in Table 1 correspond to the following:

1. Sod shock tube: the canonical shock tube problem with rarefaction-contact-shock structure; while not a challenging problem, it quickly identifies algorithmic problems resolving basic wave structure.
2. Einfeldt (or 1-2-3) problem: consists of two strong rarefaction waves, with a near-vacuum between them; methods that conserve total energy might show internal energy errors for this problem.
3. Stationary contact problem: consists of a strong shock wave moving to the right, a stationary contact, and a strong rarefaction moving to the left; it is based on the left part of the well-known Woodward-Colella problem, but with the velocity shifted to make the contact stationary, and tests an algorithm's dissipation by how much the contact is smeared.
4. Slow shock problem: consists of a Mach 3 shock wave moving slowly to the right; some numerical methods exhibit unphysical oscillations behind the shock.
5. Shock-contact-shock problem: when two shocks separate from the initial state, with a contact between them, errors are produced in all fields, and this problem tests how well an algorithm deals with those errors; this is similar to the planar Noh problem but with weaker shocks.
6. LeBlanc problem: a strong shock, strong rarefaction version of the basic rarefaction-contact-shock problem; it is a good test of a method's robustness.

Table 1: Parameters for the six 1D Riemann problems.

<table>
<thead>
<tr>
<th>Test</th>
<th>( x_{\text{int}} ) [cm]</th>
<th>( u_{\text{in}} ) [s]</th>
<th>( \gamma ) [-]</th>
<th>( \rho_L ) [g/cm(^3)]</th>
<th>( u_L ) [cm/s]</th>
<th>( p_L ) [dyn/cm(^2)]</th>
<th>( \rho_R ) [g/cm(^3)]</th>
<th>( u_R ) [cm/s]</th>
<th>( p_R ) [dyn/cm(^2)]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5</td>
<td>0.25</td>
<td>7/5</td>
<td>1.0</td>
<td>0.0</td>
<td>1.0</td>
<td>0.125</td>
<td>0.0</td>
<td>0.1</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0.15</td>
<td>7/5</td>
<td>1.0</td>
<td>-2.0</td>
<td>0.4</td>
<td>1.0</td>
<td>2.0</td>
<td>0.4</td>
</tr>
<tr>
<td>3</td>
<td>0.8</td>
<td>0.012</td>
<td>7/5</td>
<td>1.0</td>
<td>-19.59745</td>
<td>10 (^3)</td>
<td>1.0</td>
<td>-19.59745</td>
<td>10 (^{-2})</td>
</tr>
<tr>
<td>4</td>
<td>0.5</td>
<td>1.0</td>
<td>7/5</td>
<td>3.857143</td>
<td>-0.810631</td>
<td>10.33333</td>
<td>1.0</td>
<td>-3.44</td>
<td>1.0</td>
</tr>
<tr>
<td>5</td>
<td>0.5</td>
<td>0.3</td>
<td>7/5</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
<td>1.25</td>
<td>-0.5</td>
<td>1.0</td>
</tr>
<tr>
<td>6</td>
<td>0.3</td>
<td>0.5</td>
<td>5/3</td>
<td>1.0</td>
<td>0.0</td>
<td>(2/3)(\times)10 (^{-1})</td>
<td>10 (^{-2})</td>
<td>0.0</td>
<td>(2/3)(\times)10 (^{-10})</td>
</tr>
</tbody>
</table>
Mesh: For all problems, $x_{\text{min}} = 0.0$ and $x_{\text{max}} = 1.0$ cm; in 2D, $y_{\text{min}} = 0.0$, $y_{\text{max}} = 0.2$ cm.
1D: $N_x = 50, 100, 200, 400, 800$  
2D: $N_x, N_y = (50, 10), (100, 20), (200, 40), (400, 80), (800, 160)$
1D: $\Delta x = 0.02, 0.01, 0.005, 0.0025, 0.00125$  
2D: $\Delta x = \Delta y = 0.02, 0.01, 0.005, 0.0025, 0.00125$

Initial conditions: Uniform and constant material density, pressure, and velocity on each side of the initial interface ($x = x_{\text{int}}$). Other states are related through the polytropic equation of state: $p = (\gamma - 1)\rho e$ where $\gamma$ is the (constant) ratio of specific heats.

Boundary conditions: The problem is to be run so that any spurious waves generated by boundary conditions do not affect the solution on the Comparison Domain, defined below. Possible boundary conditions are:
- Left/Right: $x_{\text{min}}, x_{\text{max}}$: Constant in time
- Top/Bottom: $y_{\text{min}}, y_{\text{max}}$: Reflective

Output, in ASCII comma- or space-delimited format*, to include:
- i. Values of density, velocity, pressure, specific internal energy as a function of position at $t_0$ and $t_{\text{fin}}$.
- ii. Entire mesh total energy, kinetic energy, internal energy as a function of time.

Comparison Domain:  
1D: $x \in [0, 1]$  
2D: $(x, y) \in [0, 1] \times [0, 0.2]$

* Sample output available upon request.
1D Riemann Problem Results

Figure 1. Riemann #1 results.

Figure 2. Riemann #2 results.

Figure 3. Riemann #3 results.

Figure 4. Riemann #4 results.

Figure 5. Riemann #5 results.

Figure 6. Riemann #6 results.