RELAP5-3D Verification 2014

Dr. George Mesina

RELAP5 International Users Seminar
Idaho Falls, ID
September 11-12, 2014
Outline

- Verification status in 2013
- Progress on code issues revealed by verification
- New Verification testing
- New Verification capability
Verification Capability: Summary

- Automated verification capability introduced in 2013 to detect code errors (detection) for many code features (coverage)

- DETECTION: RELAP5-3D writes a verification file that records primary variable sums to 32+ decimal place

- COVERAGE: The verification test suite has
  - Tests 194 code features
  - 43 test problems with 125 input cases

- Comparing verification files for the same input reveals changes between code versions or application of code capability

- THEOREM: Verification Comparison NEVER finds differences falsely.
  - Provided it is programmed correctly.
## COVERAGE: Summary of Code Features Tested

<table>
<thead>
<tr>
<th>Feature Category</th>
<th>Number of features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hydro component</td>
<td>29</td>
</tr>
<tr>
<td>Volume flag - tlpvbfe</td>
<td>7</td>
</tr>
<tr>
<td>Wall friction options</td>
<td>6</td>
</tr>
<tr>
<td>Junction flag - jefvcahs</td>
<td>14</td>
</tr>
<tr>
<td>Junction form loss</td>
<td>4</td>
</tr>
<tr>
<td>Flow regimes</td>
<td>6</td>
</tr>
<tr>
<td>Heat structure type</td>
<td>3</td>
</tr>
<tr>
<td>Heat transfer modes</td>
<td>7</td>
</tr>
<tr>
<td>Heat structure BC types</td>
<td>8</td>
</tr>
<tr>
<td>Heat source options</td>
<td>5</td>
</tr>
<tr>
<td>Material Prop</td>
<td>3</td>
</tr>
<tr>
<td>Metal-Water</td>
<td>3</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td><strong>95</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Feature Category</th>
<th>Number of features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enclosure</td>
<td>2</td>
</tr>
<tr>
<td>Reactor kinetics</td>
<td>12</td>
</tr>
<tr>
<td>Decay Heat</td>
<td>11</td>
</tr>
<tr>
<td>Trips</td>
<td>2</td>
</tr>
<tr>
<td>Control variables</td>
<td>32</td>
</tr>
<tr>
<td>Tables</td>
<td>8</td>
</tr>
<tr>
<td>Flow regimes</td>
<td>6</td>
</tr>
<tr>
<td>Equation Solvers</td>
<td>5</td>
</tr>
<tr>
<td>Card 1 Options</td>
<td>9</td>
</tr>
<tr>
<td>Proprietary</td>
<td>5</td>
</tr>
<tr>
<td>Other Major Options</td>
<td>7</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td><strong>99</strong></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>194</strong></td>
</tr>
</tbody>
</table>
### DETECTION: Verification File

**RELAP5-3D/Ver: 4.1.3 steelers.inl.gov**

- **Time compiled:** Aug 14 2013 13:29:15
- **Date and Time of run:** 13/08/14 15:04:49

#### Case 1: edward's pipe problem base case with extras

<table>
<thead>
<tr>
<th>Dump</th>
<th>1</th>
<th>Advancement=</th>
<th>109</th>
<th>time=</th>
<th>1.0000E-01</th>
</tr>
</thead>
<tbody>
<tr>
<td>P=</td>
<td>4.9365983737086219E+07</td>
<td>401878A1E58D75B0000000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Uf=</td>
<td>1.964950740408072E+07</td>
<td>401723E3E7AFC05FEC00000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ug=</td>
<td>5.45204898553964E+07</td>
<td>40199F55EB2E60AE0000000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOIDg=</td>
<td>7.015848922104199E+00</td>
<td>4001C103AB179E074A00000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QUALa=</td>
<td>0.0000000000000000E+00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boron=</td>
<td>0.0000000000000000E+00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vf=</td>
<td>2.044821329728118E+02</td>
<td>400698F6DA1FDA3236D4000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vg=</td>
<td>2.3165076689908255E+02</td>
<td>4006CF4D3151A9C1FEC1000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **P=** 4.9365983737086219E+07
- **Uf=** 1.964950740408072E+07
- **Ug=** 5.45204898553964E+07
- **Vf=** 2.044821329728118E+02
- **Vg=** 2.3165076689908255E+02

<table>
<thead>
<tr>
<th>Case 2</th>
<th>2</th>
<th>Advancement=</th>
<th>509</th>
<th>time=</th>
<th>5.0000E-01</th>
</tr>
</thead>
<tbody>
<tr>
<td>P=</td>
<td>1.1610017826711973E+07</td>
<td>4016624F43A746CAAC00000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Uf=</td>
<td>1.3706563288757732E+07</td>
<td>4016A24A8693D80DB180000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ug=</td>
<td>5.3792556235069888E+07</td>
<td>40189A67961E0C000000000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VOIDg=</td>
<td>2.127747744316551E+00</td>
<td>4003420B4137FFA341800000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QUALa=</td>
<td>0.0000000000000000E+00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boron=</td>
<td>0.0000000000000000E+00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vf=</td>
<td>2.8891214895206032E+02</td>
<td>400720E98297E02E048B000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vg=</td>
<td>9.1675057057563033E+02</td>
<td>4008CA6012652E840000000000000000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **CPU Time:** 3.609449999999996E-01 hour
- **Bytes (Upper Limit):** 2764
Verification Capability: 2013 Summary

- Automated verification capability introduced in 2013.
  - Tests 194 code features via 43 test problems, with 125 input cases
  - Records primary variable sums to 32+ decimal places
- Findings based on comparing two verification files
  - Null test compares between two versions
  - Restart compares base run and run restarted from middle
  - Backup compares base run to one that repeats every timestep
- Failure means that at least on sum was not the same.

<table>
<thead>
<tr>
<th>Version 4.1.3</th>
<th>Failures in 43 Test Problems</th>
<th>Failures in 125 cases</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null Testing</td>
<td>6/43</td>
<td>6/125</td>
</tr>
<tr>
<td>Restart Testing</td>
<td>25/43</td>
<td>52/125</td>
</tr>
<tr>
<td>Backup Testing</td>
<td>37/43</td>
<td>62/125</td>
</tr>
</tbody>
</table>
Verification Capability Revealed Issues

• Inexact Restart
  – Half caused by failure to write required data to restart file.
    • Some recent modules did not have restart read/write routines.
    – Other half caused by a difference in final bit of the cumulative time.
      • Resolved by updating certain “integer time” information at edit points rather than at timecard end times only

• Example
  – ANS79 has 9 cases. Four failed that represented more than one nuclide for decay heat.
  – The variable that recorded that was not written to the restart file.
Verification Capability Restart Issues

• Other half caused by a difference in final bit of the cumulative time.
• The issue was that the cumulative time did not match across a restart.
  – TIMEHY = “Start of timecard” + “integer time”*dt_{small}.
  – On a restart, there were two situations
    • From end of a timecard, integer time was zero
    • From middle of timecard it was non-zero
    • For large integers, the product could be off in the final bit due to the number of digits involved
• Solution was to “recalibrate at edit times” by changing to:
  – TIMEHY = “Prior edit” + (“integer time” – “integer edit time”)*dt_{small}
  – This solved the verification restart issue
• However, it caused three PVM installation problems to fail
Verification Capability Revealed Issues

• The PVM problems that failed were synchronously coupled.
• The solution was to introduce the same recalibration:
  – In the PVM DTSTEP subroutine
  – In the related section of RELAP DTSTEP
  – This solved the PVM installation problem issue while keeping the verification restart solution
Verification Capability Revealed Issues

- Inexact Backup
  - Most caused by failure to record an old-time value of some important quantity, such as QUALE.
    - Resolved by adding an old-time quantity to the relevant module data (usually a derived type).
  - Most changes occur in subroutine MOVER which restores data to previous timestep values on a backup.
    - Not all were in MOVER. Some were in TRAN, HYDRO, and DTSTEP.
  - Changes to VERFBACKUP were necessary to resolve an issue with the repeat-count sums.
Verification Capability Progress

- Progress on original 43 Test Problems

<table>
<thead>
<tr>
<th>Test Case</th>
<th>4.1.3 Failures</th>
<th>4.2.1 Failures</th>
<th>4.3.1 Failures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null Test</td>
<td>6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Restart Test</td>
<td>25</td>
<td>18</td>
<td>0</td>
</tr>
<tr>
<td>Backup Test</td>
<td>37</td>
<td>20</td>
<td>0</td>
</tr>
</tbody>
</table>

- Improvements for correcting the verification issues do NOT cause INL standard test cases to fail.
New Verification Capability and Modifications

• Added verification capability
  – New input decks added
  – Multi-case input decks
  – PC verification
  – Input modification
New PVM Verification Problems Supplied

• New non-PVM input decks added
  – Increase coverage
• New PVM Coupling Problems
  – Backup issues
  – Restart issues
• These required expansion of the Verification Directory and Makefiles
  – Each new problem type has its own subdirectory with Make.inp
  – The subdirectory Makefile now has PVM null, restart, and backup targets
  – A new Input variable for PVM tests added to Make.tests
  – The main Makefile can run all PVM null, restart, and backup tests
New PVM Verification Problems Supplied

- Backup issues had a variety of causes, including:
  - Backups cannot proceed from the first step after an explicit coupling exchange with a leader-follower.
    - Had to postpone backups till the second advancement
  - TestMatrixDT coding was triggered by setting the air appearance flag. This was changed to setting Isuces = -1.
    - This allows a section of coding in subroutine MOVER to be exercised

- Some restart problems fail because the time step does not match.
  - One fix is resetting integer time, ITIME, on restart
  - Another is converting integers to real in quadruple precision

- These problems are being worked.
Added Verification Capability

- Multi-case input deck testing
  - User found the code failed in a multi-case input, but when run as two separate input decks, both decks ran
  - Questioned whether the code produced the same calculations in multi-case form.

- Test with verification capability
  - Script written to break an input deck with N cases into N pairs of input decks
  - E.G. for deck.i with N input cases, these are named
  - Deck.cK.i collapses the first K cases into one by eliminating the “slash” input case separator
  - Deck.sK.i runs the first K cases separately by keeping slashes
  - The runs are compared
Added Verification Capability

- Only two of the 43 verification test suit input problems show differences
- PC verification Makefiles
  - The Makefiles that run the verification test suite for Linux have been rewritten to run on the Windows 7 PC.
  - There are some differences in the way the DOS nmake utility operates compared with Linux make.
- Running showed that RELAP5-3D/version 4.1.3 had the same performance on Windows 7 as Linux
Change to 199 Card Verification Control Card

- The 199 activates verification
  - 199  Word(1)  Word(2)  Word(3)  Word(4)
  - Word(1) can be “verify” or “noverify”
  - Word(3) = start, integer advancement or real time
  - Word(4) = shut off-advancement control (integer)

- Word(2) values will change. In 4.2.1 these are the values:
  - dump – write verify dumps on specified steps
  - backair – backup for air (non-condensable) appearance
  - backpck – backup for water packing
  - backvel – backup for velocity flip-flop
  - backall – backup every timestep, 2 forward / 1 back

- Change: Starting in 4.3.1, eliminate backpck, backvel, and backair
SUMMARY

• The verification capability is being used to locate code problems with:
  – Unexpected calculation changes going from version to version
  – Restart issues
  – Backup issues
  – Multi-case issues
  – PVM coupling issues

• All issues uncovered with the original Verification test suit problems have been solved (as of version 4.3.1).

• New issues have been reported with multi-case and PVM coupling. These are being worked

• New verification capability has been developed.