Jacobian Consistency Tool for RELAP5-3D

Dr. George L Mesina, INL

2018 RELAP5 International Users Seminar
April 18-19, 2019
Idaho Falls

INL/CON-19-53510

Outline

- Background
- Jacobian Approximated well?
- Compare approximations of Jacobians
- J-Tool to generate & compare Jacobians
- J-Tool Input, studies, and results
- Conclusions

Background: Conservation Equations

1. $$\frac{\partial(\alpha_g \rho_g X_n)}{\partial t} + \frac{1}{A}\frac{\partial}{\partial x}(\alpha_g \rho_g X_n v_g A) = 0$$ Noncondensable Mass

2. $$\frac{\partial(\alpha_g \rho_g U_g)}{\partial t} + \frac{1}{A}\frac{\partial \alpha_g \rho_g U_g v_g A}{\partial x} = -P\frac{\partial \alpha_g}{\partial t} - \frac{P}{A}\frac{\partial(\alpha_g v_g A)}{\partial x} + Q_{wg} + Q_{ig} - Q_{gf} + \Gamma_{ig} h_g^* + \Gamma_w h_g' + DISS_g$$ Gas Energy

3. $$\frac{\partial(\alpha_f \rho_f U_f)}{\partial t} + \frac{1}{A}\frac{\partial \alpha_f \rho_f U_f v_f A}{\partial x} = -P\frac{\partial \alpha_f}{\partial t} - \frac{P}{A}\frac{\partial(\alpha_f v_f A)}{\partial x} + Q_{wf} + Q_{if} + Q_{gf} - \Gamma_{ig} h_f^* - \Gamma_w h_f' + DISS_f$$ Liquid Energy

4. $$\frac{\partial(\alpha_g \rho_g)}{\partial t} - \frac{\partial(\alpha_f \rho_f)}{\partial t} + \frac{1}{A}\frac{\partial(\alpha_g \rho_g v_g A - \alpha_f \rho_f v_f A)}{\partial x} = 2\Gamma_g$$ Diff Continuity

5. $$\frac{\partial(\alpha_f \rho_f)}{\partial t} + \frac{\partial(\alpha_f \rho_f)}{\partial t} + \frac{1}{A}\frac{\partial(\alpha_g \rho_g v_g A + \alpha_f \rho_f v_f A)}{\partial x} = 0$$ Sum Continuity

Background

- The Primary variables are $\bar{x}$ and $\bar{v}$ where:
- The Conserved quantities, $\bar{F}(\bar{x})$, are on the left side of the **conservation** equations

$$\bar{v} = \begin{bmatrix} v_f \\ v_g \end{bmatrix} \qquad \bar{x} = \begin{bmatrix} X_n \\ U_g \\ U_f \\ \alpha_g \\ P \end{bmatrix} \qquad \bar{F}(\bar{x}) = \begin{bmatrix} \alpha_g \rho_g X_n \\ \alpha_g \rho_g U_g \\ \alpha_f \rho_f U_f \\ \alpha_g \rho_g - \alpha_f \rho_f \\ \alpha_f \rho_f + \alpha_f \rho_f \end{bmatrix}$$

- RELAP5 **adapts** the equations for computer solution as follows:
  - Insert correlations for mass transfer, Γ, and energy transfer, Q's
  - Expand derivatives with multiplication and chain rule
  - Discretize in time and space
  - Replace new time temperatures & densities with 1st order multi-dimensional Taylor Polynomials
- **Result** is a linear system in the primary variables in Vol. 1, Sec. 3.1.4
  - Designate the coefficient matrix as the "**Analytical**" **Jacobian**, J_A.
  - $J_A \frac{\partial \tilde{x}}{\partial t} = \bar{b}_A^n + \overline{B_A}(\bar{v}^{n+1})$

Solution. How good is it?

- So: $\left(\frac{\Delta \bar{x}}{\Delta t}\right)^{n+1} = J_A^{-1}\bar{b}_A^n + J_A^{-1}\overline{B_A}(\bar{v}^{n+1})$
- **Time advancement**: $\bar{x}^{n+1} = \bar{x}^n + \Delta t \left(\frac{\Delta \bar{x}}{\Delta t}\right)^{n+1}$

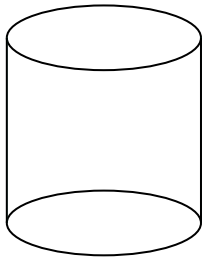
Velocity Solution

- To get $\bar{v}^{n+1}$, do same kinds of operations to the **momentum equations**. Have only pressure and velocity at new time, n+1.
  - $J_M \left(\frac{\Delta \bar{v}}{\Delta t}\right)^{n+1} = \bar{b}_v^n + B_v \left(\frac{\Delta \bar{P}}{\Delta t}\right)^{n+1}$
  - Solve $J_A \frac{\partial \tilde{x}}{\partial t} = \bar{b}_A^n + \overline{B_A}(\bar{v}^{n+1})$ to get $\left(\frac{\Delta \bar{P}}{\Delta t}\right)^{n+1}$
  - Substitute $\left(\frac{\Delta \bar{v}}{\Delta t}\right)^{n+1} = J_M^{-1}\bar{b}_v^n + J_M^{-1}B_v \left(\frac{\Delta \bar{P}}{\Delta t}\right)^{n+1}$
  - $\bar{v}^{n+1} = \bar{v}^n + \Delta t \left(\frac{\Delta \bar{v}}{\Delta t}\right)^{n+1}$

- **How well does J_A approximate the actual Jacobian?**
- **Compare: Analytical Jacobian, J_A, with Numerical Jacobian, J_N.**

Discrete Model Used as Basis for Comparison

- To build a Jacobian Matrix with RELAP5-3D coding, need a model and input values.
- System = a single control volume
- Jacobian J_A (or J_N) involves the mass and energy equations only, not the momentum equations
  - No trips, controls, neutronics
  - Explicit coupling to heat conduction
- Geometry – Control Volume = SNGLVOL
  - Volume 1m^3, dL=1m, Area=1m^2.
  - No junctions
- Semi-implicit advancement, dt = 1s
- Fluid = H2ON



Representative Control Volume

Calculate Analytical and Numerical Jacobian

- Input values for $\overline{x}^n$
  - Actually requires over 100 quantities initialized
- Build analytical Jacobian J_A
  - Use formulation of Vol. 1, Section 3.1.4
  - Call RELAP5-3D subroutines (STATEP, VEXPLT, PRESEQ)
  - Separates temporal, mass transfer, and energy transfer portions
- Build numerical Jacobian, J_N
  - Loop over elements of $\overline{x} = (x_1, x_2, x_3, x_4, x_5) = (X_n, U_g, U_f, \alpha_g, P)$
  - Find Forward = $\frac{\overline{F}_j - \overline{F}_j(\overline{x})}{\Delta \overline{x}_k}$ or Backward = $\frac{\overline{F}_j(\overline{x}) - \overline{F}_j(\overline{x} - \Delta \overline{x}_k)}{\Delta \overline{x}_k}$ diference, $1 \le j, k \le 5$
    - Use backward whenever forward involves one single-phase and one two-phase fluid state or one stable and one meta-stable (or worse)
  - Use STATEP to get properties at $(\overline{x} + \Delta \overline{x}_k)$ or $(\overline{x} - \Delta \overline{x}_k)$

Decomposition for Detailed Jacobian Study

- Jacobian, J, is broken into 3 submatrices
  - J^{m} = Derivs of **mass transfer** terms (Diff Mass Eqn => Matrix row 4 only)
  - J^{e} = Derivs of **energy transfer** terms (Energy Eqn => Matrix row 2 and 3)
  - $J^{t} = J - J^{m} - J^{e}$ = Derivatives of **temporal** terms
- Mass xfer term: $\Gamma_{ig} = -\frac{\frac{P_s}{P}H_{ig}\left[T^s(P_s)-T_g\right]+H_{if}\left[T^s(P_s)-T_f\right]}{h_g^*-h_f^*}$
  - Mass transfer occurs in difference equation only

$$J^m = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 2\frac{\partial \Gamma_{ig}}{\partial X_n} & 2\frac{\partial \Gamma_{ig}}{\partial U_g} & 2\frac{\partial \Gamma_{ig}}{\partial U_f} & 0 & 2\frac{\partial \Gamma_{ig}}{\partial P} \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- In eqn. 2, $Q_2 = -Q_{ig} + Q_{gf} - \Gamma_{ig} h_g^*$ is energy transfer term
- In eqn. 3, $Q_3 = -\mathrm{Q_{if}} - \mathrm{Q_{gf}} + \Gamma_{ig}\mathrm{h_f^*}$. $\boldsymbol{Q_3 = -Q_2}$
  - $\mathrm{Q_{if}} = \mathrm{H_{if}}(\mathrm{T^s}(\mathrm{P_s}) - \mathrm{T_f})$
  - $\mathrm{Q_{ig}} = \frac{\mathrm{P_{s,L}}}{\mathrm{P_L}}\mathrm{H_{ig}}\left(\mathrm{T^s}(\mathrm{P_s}) - \mathrm{T_g}\right)$
  - $\mathrm{Q_{gf}} = \left(\frac{\mathrm{P}-\mathrm{P_s}}{\mathrm{P}}\right)\mathrm{H_{gf}}\left(\mathrm{T_g} - \mathrm{T_f}\right)$

$$J^e = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \bar{Q}_2}{\partial X_n} & \frac{\partial \bar{Q}_2}{\partial U_g} & \frac{\partial \bar{Q}_2}{\partial U_f} & \frac{\partial \bar{Q}_2}{\partial \alpha_g} & \frac{\partial \bar{Q}_2}{\partial P} \\ \frac{\partial \bar{Q}_3}{\partial X_n} & \frac{\partial \bar{Q}_3}{\partial U_g} & \frac{\partial \bar{Q}_3}{\partial U_f} & \frac{\partial \bar{Q}_3}{\partial \alpha_g} & \frac{\partial \bar{Q}_3}{\partial P} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- Apply to both Analytical and Numerical Jacobian

Jacobian Tool High-level Algorithm Description

Loop over (thousands of) input fluid states

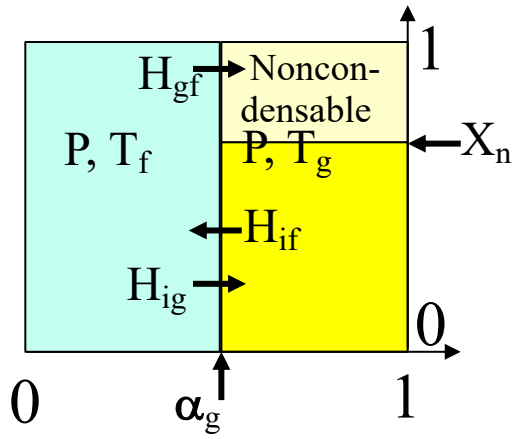
- Read an input fluid state
  - Initialize and calculate all RELAP5-3D data needed to build a Jacobian Matrix in subroutine PRESEQ
- Build Analytical & num. Jacobian submatrices: J_A^t, J_A^e, J_A^m and J_N^t, J_N^e, J_N^m
- Calculate how close Jacobians are
  - All pairwise differences of elements in corresponding submatrices
- Calculate condition number J_A or J_N
- Output differences & condition numbers for the current fluid state

End Loop

Output summary of worst conditions among all input states

Initialization of Jacobian Tool Data

- Input specifies the fluid state to examine, 11 quantities
  - A fluid state = (P, α_g, X_n, T_g, T_f)
    - P = Pressure, α_g = void fraction
    - X_n = noncondensable quality
    - T_g, T_f = Temperature of gas and liquid phases
  - Indicator Flag: Absolute or "from saturation"
  - Heat transfer coefficients: H_{ig}, H_{if}, H_{gf}
  - Old time quantities: $\alpha_{g,old}$, $X_{n,old}$



- Convert T_g and T_f to internal energy, U_g and U_f
  - Find partial pressure of steam. Call TSTATE & STATEP
- Derived RELAP5-3D data must be set or built
  - Much module data: volume, list vectors, system, components, junction, control, auxiliaries, etc. About 100 quantities in all
- After input and initialization, calculate rest of properties from STATEP

J-Tool Features

- Accept internal energies or Temperatures (convert to internal energy)
- Can accept absolute or relative (to saturation) temperatures
  - Calculates T^{sat} from input P and X_n.
- Recognizes and rejects many invalid states
- Calculates and stores Jacobian Relative Errors across all N states
  - Relerr_**t**(i, j, k) = $|J^{t}_{A,i,j} - J^{t}_{N,i,j}| \,/\, \max(J^{t}_{A,i,j}, J^{t}_{N,i,j})$
  - Relerr_**m**(j, k) = $|J^{m}_{A,4,j} - J^{m}_{N,4,j}| \,/\, \max(J^{m}_{A,4,j}, J^{m}_{N,4,j})$
  - Relerr_**e**(j, k) = $|J^{e}_{A,2,j} - J^{e}_{N,2,j}| \,/\, \max(J^{e}_{A,2,j}, J^{e}_{N,2,j})$
  - MaxRelerr_**t**(i, j) = max {Relerr_**t**(i, j, k), k=1, … ,N}
  - MaxRelerr_**m**(j) = max {Relerr_**m**(j, k), k=1, … ,N}
  - MaxRelerr_**e**(j) = max {Relerr_**e**(j, k), k=1, … ,N}
- Uses LAPACK to calculate condition numbers of $J = J^{t} + J^{m} + J^{e}$
  - For both Analytical and Numerical Jacobian

Jacobian New Modules

- Jacbldmod – subroutines for calculating the analytical Jacobian
  - Original way and in submatrix pieces
  - Portions of vexplt.F
  - Corrected flexible wall
  - void ramp subroutine
- Jacobmod – subroutines for Jacobian Tool
  - File opening (Jacobian input & output, Water properties)
  - Input reading, output: header, case, final
  - Initialization, permanent and per case
  - Numerical Jacobian calculations
  - Statistics collection & condition number calculation

Major J-Tool Subroutines and files

- Preseq (relap/ directory)
  - Portions moved into jacbldmod, option to build Jacobian 2 ways
- Istate (relap/ directory)
  - Modified to produce same values from J-Tool as from RELAP5-3D
- Jacprintprop – new (relap/ directory)
  - Various Diagnostic output of variables used to construct Jacobian
- Jacobian Directory
  - Jacobian – main program
  - Open/Initialize subroutines – jacoballor5, jacobfluid / initvlm, jacinitfprp
  - State Input & Calc Data – jacreadstate
  - Jacobian calc – jacnum, temporalterms
  - Diagnostics – jacobdiag, jacprintprop, unitcheck, and others…
  - Scripts – Makefile, input_gen2.py
  - Input – jacob_2.in

Performing Studies

- Python script, input_gen2.py, generates input files for studies. E.G.

```
voids = [0.001, 0.1, 0.5, 0.9, 0.999]
tempf = [-50.0, -10.0, -1.0, -0.1, 0.0, 1.0, 10.0]
tempg = [-5.0, -1.0, 0.0, 1.0, 100.0, 500.0]
Hig = 2.090156593143735E+04
Hif = 6.92691E-02
Hig = 5.65530E+05
flag = 1.0
mult = 3000/14.7
spacing = math.log(mult)/6
for press in range(7):
  p = 6894.75729*14.7*math.exp(press*spacing)
  for voidg in voids:
    voidgo = voidg*0.9
# NC-Quality. There can be no NC but NEVER ONLY NC.
   for qual in qualities:
      quala = qual
      if voidg == 0.0:
        quala = 0.0
      qualao = quala*0.9
      for tf in tempf:
        for tg in tempg:
# Output State N-Tuple (state point)
          print p, voidg, quala, tg, tf, Hgf, Hif, Hig, flag, voidgo, qualao
```

J-Tool Discoveries

- Study 1 was comprised of 10290 fluid states
- Condition Numbers of the 5x5 Jacobian Matrices are LARGE
  - Maximum was O(10^{21})
  - Minimum was O(10^{5})
  - Many of the relative errors were above tolerance of 5%
- Study 2 of 7350 combinations of primary variables

	P	α_g	X_n	ΔT_f	ΔT_g	H_{if}	H_{ig}	H_{gf}
1	101352.932163	0.001	0.0	-50.0	-5.0	$6.927e-2$	5.6553e5	1.7587e4
2	245925.542130	0.1	0.1	-10.0	-1.0			
3	596720.499166	0.5	0.5	-1.0	1.0			
4	1447899.030900	0.9	0.8	0.0	100.0			
5	3513222.030440	0.999	0.99	1.0	500.0			
6	8524578.559510			10.0				
7	20684271.870000							

J-Tool Initial Results

- **Study 1** had case with Tf = 0 C and Xn = 1.0.
- **Study 2,** 7350 cases, eliminated those cases.
  - Relative errors shown in table. Red indicates above tolerance

Jacobian Temporal Term, J^t, Relative Errors

$\partial/\partial X_n$	$\partial/\partial U_g$	$\partial/\partial U_f$	$\partial/\partial \alpha_g$	$\partial/\partial P$
1.28E-01	1.66E-04	0	1.00E+00	9.30E-06
1.00E+00	1.20E+00	0	2.97E-13	9.43E-01
0	0	9.73E-02	1.91E-10	1.12E-01
1.00E+00	7.47E-04	8.11E-04	2.59E-10	6.97E-05
1.00E+00	7.47E-04	8.11E-04	2.39E-08	6.97E-05

Jacobian Mass Transfer, J^m, Relative Errors

1.98E+00	1.00E+00	1.00E+00	0	1.86E+00

Jacobian Energy Transfer, J^e, Relative Errors

1.86E+00	1.00E+00	1.00E+00	0	1.85E+00

- To find causes, undertook new studies and created many diagnostics to zero in on sources.

J-Tool Discoveries

- **Study3**: Worst error inducing state was case 3178
- P =1447899.0309, $\alpha_g = 0.001$, X_n= 0.0, ΔT_f **= 0.0**, ΔT_g = +1.0.
- Discovered that for $T_f = T^{sat}$ to 17 decimal places, calculated liquid density and enthalpy in the bulk fluid are very close to the gas values. **Error**.
- To Test
  - Modified RELAP5-3D input to accept 17 decimal places
  - Created input file jvol.i with $T_f = T^{sat}$ to 17 decimal places
- **RELAP5-3D produced same calculations as J-Tool**

- **Study 4**: Heat Transfer Coefficients
  - Reduced rel. errors using Heat Transfer Coefficients from a RELAP5 run
  - Reduced rel. errors Heat Xfer Coefficients from time zero initialization
  - Both reduced many errors further

J-Tool Discoveries

- **Study 5**: Avoiding unstable fluids with H2ON
  - Worst remaining case had **energy** transfer *relative error* ***> 1***
    - $J^{e}_{A,2,j}$ and $J^{e}_{N,2,j}$ had **opposite signs** for j = 1 or j = 5.
  - Occurred at **P = 20,684,271.87 MPa**, $\alpha_g = 0.001$, X_n = 0.0, ΔT_f = -50.0C, and ΔT_g **= -5.0C**
    - $T^s == 641.76 == T_g = 50 + T_f$ to 14 decimal places.
    - But ΔT_g = -5.0 should produce $T_g = 636.76$.
  - Input file jvol4.i with extra diagnostics shows that RELAP5-3D reads in $T_g = 636.76$, calculates $T^s = 641.76$, but resets $T_g = T^s = 641.76$ after the time zero edit in subroutine STATEP in the nh5 section.
- Avoid unstable fluids at high pressure. Use top pressure under 19 Mpa
  - Why? Next slide…

J-Tool Discoveries

- **Study 5 continued:** Examine ASCII fluid property table. Distance from saturation through the metastable liquid to unstable appears less than 10 degrees for high pressures (above 20 MPa)
  - This does not occur when old H2O property file is used
  - It did not occur with pressures below 19 MPa with H2ON

- **Study 6:** Effect of Consistent fluid
  - RELAP5 can linearly interpolate between properties calculated:
    1. Usual way with Clausius-Clapyron Eqns. (Vol. 1 Sec. 3.2.2)
    2. New Consistent Fluid way: linearly interpolated from property table
    - Default is Clausius (ω = 1 for). Input ω = 0 for Consistent Fluid
  - Previous work showed ω = 0 (Consistent) was better more often than not by certain measures
  - As ω decreases, the relative error in the $J^t_{3,5}$ decreases, but it increases in 6 other elements of J^t, one of J^e, and one of J^m.

J-Tool Discoveries

- With the previous adjustments, the **MAX**IMUM RELATIVE Errors are
  - Avoid pathological ΔT values, especially near P_{crit}
  - Use reasonable H's, Clausius-Clapyron Eqns.

		$\Delta/\Delta X_n$	$\partial/\partial U_g$	$\partial/\partial U_f$	$\partial/\partial \alpha_g$	$\partial/\partial P$
Jt	NC qual.	10.8%	3.08E-**05**	0	2.52E-**13**	9.30E-**06**
	Gas energy	4.78%	12.5%	0	2.36E-**13**	1.47%
	Liq. energy	0	0	2.45E-**03**	1.91E-**10**	7.84E-**03**
	Diff. Mass	4.78%	3.11E-**05**	2.63E-**04**	2.39E-**10**	9.40E-**06**
	Sum Mass	4.78%	3.11E-**05**	2.63E-**04**	2.48E-**10**	9.40E-**06**
Jm	Mass Xfer	79.6%	8.20%	53.5%	0	15.2%
Je	Energy Xfer	51.0%	8.18%	53.5%	0	10.9%

- For most elements, worst case relative errors are reasonably small
- Further study is indicated for the elements not shown in black

J-Tool Summary 1

- A Jacobian Matrix tool is available for developers to analyze and identify areas for improvement of the solution process
- J-tool is a standalone program
  - No effect on standalone RELAP5-3D calculations or run speed
  - Finds Relative Errors as differences of RELAP5-3D calculations & 1st order accurate Finite Difference Approximations (FDA).
  - Reports worst case in study for each element independently
- J-Tool revealed pathological cases for RELAP5-3D
  - $T_f == T^{sat}$
  - Subcooled fluid at extreme pressures
- There is room for RELAP5-3D improvement in columns 1 and 3, i.e. derivatives w.r.t $\mathbf{X_n}$ and $\mathbf{U_f}$.

J-Tool Summary 2

- Excluding the pathological cases, analysis of WORST CASE RELATIVE errors for the temporal terms shows:
  - 18 of 25 temporal terms are below 0.25%
  - 23 of 25 are below 5%. ALL 25 are below 13%
- Excluding the pathological cases, analysis of WORST CASE RELATIVE errors for the Mass and Energy transfer terms shows:
  - 6 of 10 terms are below 16%
  - 4 of 10 need further investigation
- The relative errors calculations could be improved by using 2^{nd} order accurate FDA for comparison
- Consistent fluid does not improve relative errors
- Jacobian matrices have **HIGH condition numbers**
  - Solving Jacobian system with partial pivoting, or full pivoting, will increase accuracy