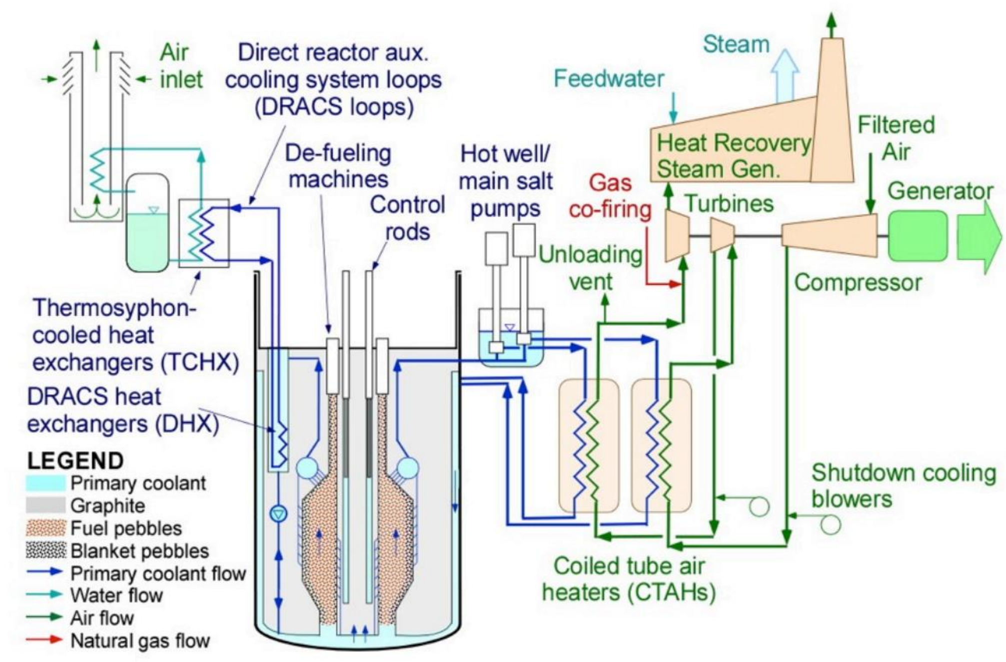
HEAT AND MASS TRANSPORT GROUP

heatandmass.ep.wisc.edu

DEPARTMENT OF Engineering Physics
UNIVERSITY OF WISCONSIN–MADISON

FIRST ITERATION 1D MODEL FOR OVERCOOLING TRANSIENTS WITH STABILIZATION USING THE SYSTEM CODE SAM AND MOOSE

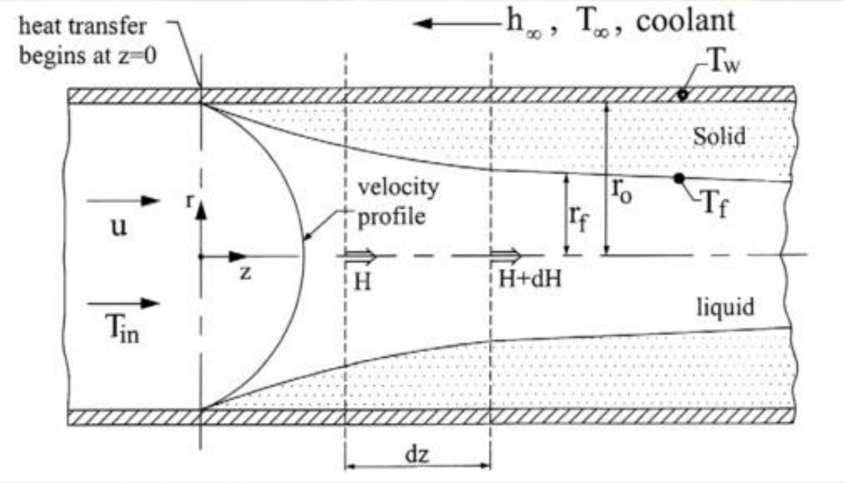
1. Background and Motivation
2. The Freezing Problem in 1D
3. Physics in MOOSE/SAM Recap
4. Issues with Numerics
5. Stabilization and Initial Tests
6. Path Forward



KAZI AHMED
NEUP Fellow
Department of Engineering Physics
University of Wisconsin - Madison
kkahmed@wisc.edu

HM Group Meeting
09 March 2018
University of Wisconsin-Madison

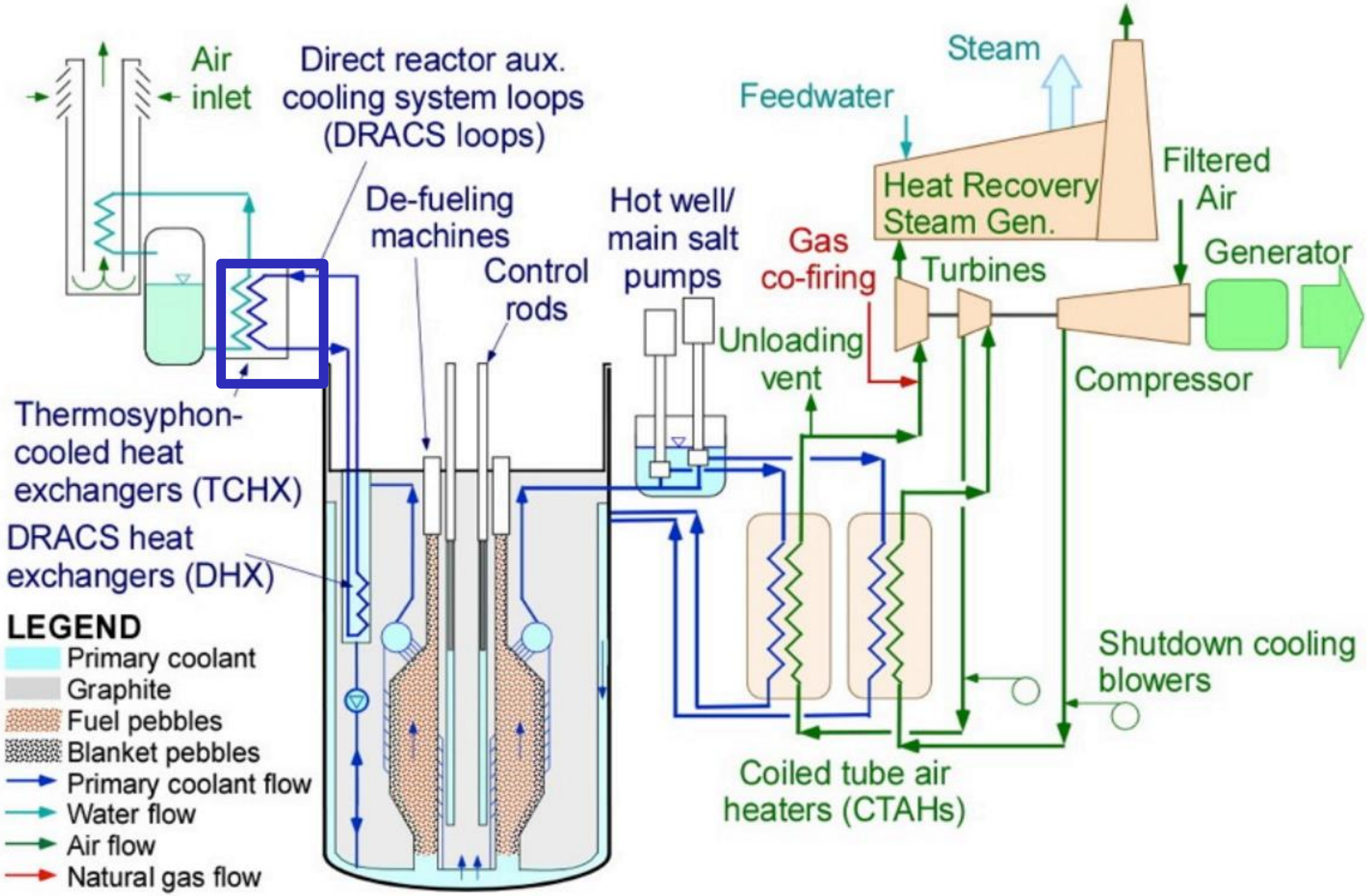
1. Background and Motivation



- *R. V. Seeniraj, G. S. Hari, "Transient Freezing of Liquids in Forced Flow Inside Convectively Cooled Tubes", International Communications in Heat and Mass Transfer 35, May 2008.*

Case Study: Mk1 PB-FHR

“Mark 1” Pebble-Bed Fluoride-Salt-Cooled High-Temperature Reactor



System Code Simulation (most basic sense)

- Network of pipes and heat structures
- Illustrates a basic component simulated in a system code

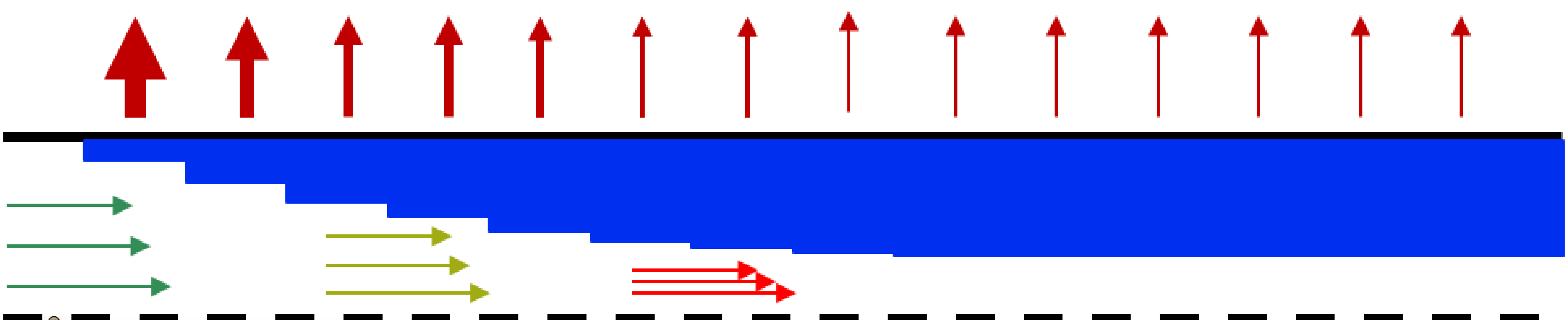
Connect inlet/outlet of pipes to build entire flow system

Flow Area, Length
Hydraulic Diameter
Heated Perimeter

Specify form losses for area changes, junctions, branches

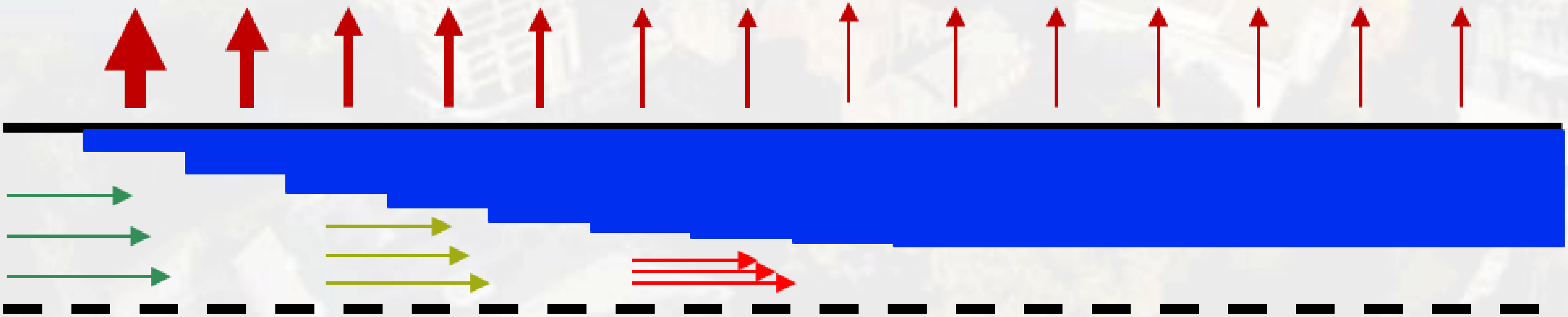
Friction loss, heat transfer coefficient:
Specified, or determined in code logic

Freezing simulation should have variable area, heat flux, etc.



DEPARTMENT OF Engineering Physics
UNIVERSITY OF WISCONSIN–MADISON

2. The Freezing Problem in 1D



1D Freezing in SAM – start from existing eq's

- For reference, the conservative governing equations from SAM are as follows

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{dz} = 0$$

$$\frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u u + p)}{dz} = -\rho g - \frac{f}{D}\frac{\rho u|u|}{2}$$

$$\frac{\partial (\rho H)}{\partial t} + \frac{\partial (\rho u H)}{dz} = \frac{q_s'' P_h}{A_c}$$

- For a freeze-capable model, start with the area-averaged generic balance with storage, convection, diffusion, body, and surface terms

$$\frac{\partial (A\rho\psi)}{\partial t} + \frac{\partial (A\rho\psi u \cdot \hat{n}_z)}{dz} + \frac{\partial (AJ \cdot \hat{n}_z)}{dz} - A\rho\phi = -\int_c J \cdot \hat{n}\, dS$$

1D Freezing in SAM – Phase equations

1. Fluid Pressure (mass)

$$\frac{\partial(\alpha\rho)_l}{\partial t}+\frac{\partial(\alpha\rho u)_l}{\partial z}=-\frac{\sigma_A}{A}$$

2. Fluid Velocity (momentum)

$$\frac{\partial(\alpha\rho u)_l}{\partial t}+\frac{\partial(\alpha\rho uu+\alpha p)_l}{\partial z}=-\rho g-\frac{f}{D}\frac{\rho u|u|}{2}$$

3. Fluid Temperature (energy)

$$\frac{\partial(\alpha\rho H)_l}{\partial t}+\frac{\partial(\alpha\rho uH)_l}{dz}=-\frac{\sigma_A H}{A_l}+\frac{q'_{wl}(T_l>T_f)}{A_l}+\frac{q'_{int}(T_l\leq T_f)}{A_l}$$

4. Solid Temperature

$$\frac{\partial(\alpha\rho H)_s}{\partial t}=\frac{\sigma_A H_s}{A}+\frac{q'_{ws}}{A}-\frac{q'_{int}}{A}$$

5. Solid Mass

$$\frac{\partial(\alpha\rho)_s}{\partial t}=\frac{\sigma_A}{A}$$

6. Interface Radius

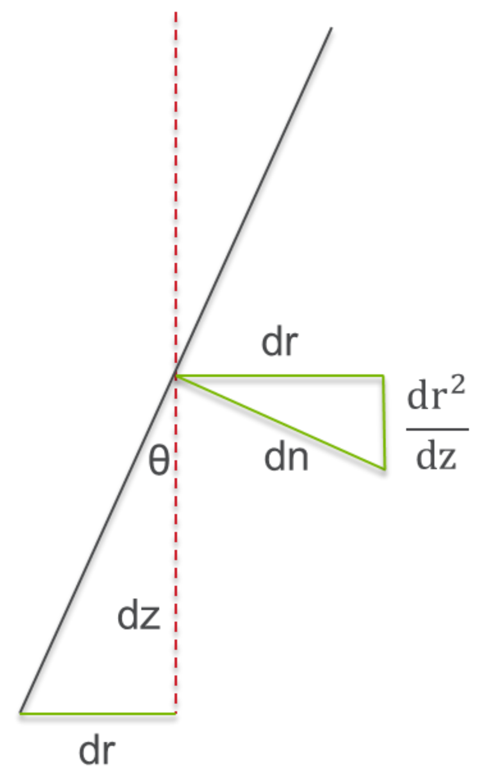
$$\left(\frac{-q''_{ws}R}{r}-h_{int}(T_m-T_l)\right)=\left(1+\left(\frac{dr_i}{dz}\right)^2\right)\frac{-\sigma_A}{\rho_s(2\pi r_i)}(\rho\Delta H)$$

Interface Radius – brief recap

- With a heat balance at the interface, the freeze front is stationary when $q_l'' = q_s''$. The interface propagates when these are not balanced

$$(q_l'' - q_s'') = \frac{\partial \hat{n}}{\partial t}(\rho \Delta H)$$

$$\left(k_s \frac{\partial T_s}{\partial \hat{n}} - k_l \frac{\partial T_l}{\partial \hat{n}}\right) = \frac{\partial \hat{n}}{\partial t}(\rho \Delta H)$$



- Use geometric relations:

$$\frac{\partial \hat{n}}{\partial t} = \frac{\partial \hat{n}}{\partial r}\frac{\partial r}{\partial t} \qquad \frac{\partial T}{\partial \hat{n}} = \frac{\partial T}{\partial r}\frac{\partial r}{\partial \hat{n}} \qquad \left(\frac{\partial r}{\partial z}\right)^2 + 1 = \left(\frac{\partial \hat{n}}{dr}\right)^2$$

- Therefore

$$\frac{\partial r}{\partial \hat{n}}\left(k_s \frac{\partial T_s}{\partial r} - k_l \frac{\partial T_l}{\partial r}\right) = \frac{\partial \hat{n}}{\partial r}\frac{\partial r}{\partial t}(\rho \Delta H)$$

$$\left(k_s \frac{\partial T_s}{\partial r_i} - k_l \frac{\partial T_l}{\partial r_i}\right) = \left(1 + \left(\frac{dr}{dz}\right)^2\right)\frac{\partial r_i}{\partial t}(\rho \Delta H)$$

Interface Radius – brief recap

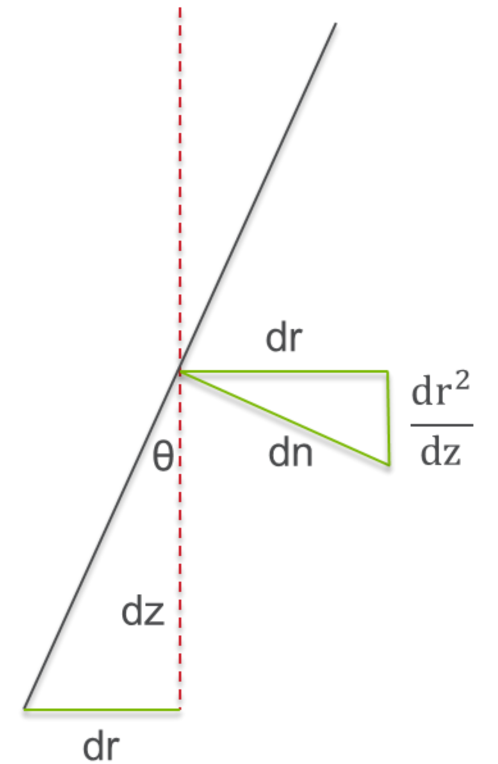
- Without a 2D simulation, one approach is to assume a fully developed profile in the solid layer

$$\frac{\partial}{\partial r}(rq) = 0 \quad \Rightarrow \quad -k_s \frac{dT_s}{dr} = \frac{C_1}{r}$$

- With the heat flux boundary condition, we have

$$C_1 = q''_{ws} R \quad \Rightarrow \quad \frac{dT_s}{dr} = \frac{-q''_{ws} R}{k_s r}$$

- If for the liquid, we assume the expression for heat transfer used earlier, the full equation to determine the rate of freezing is



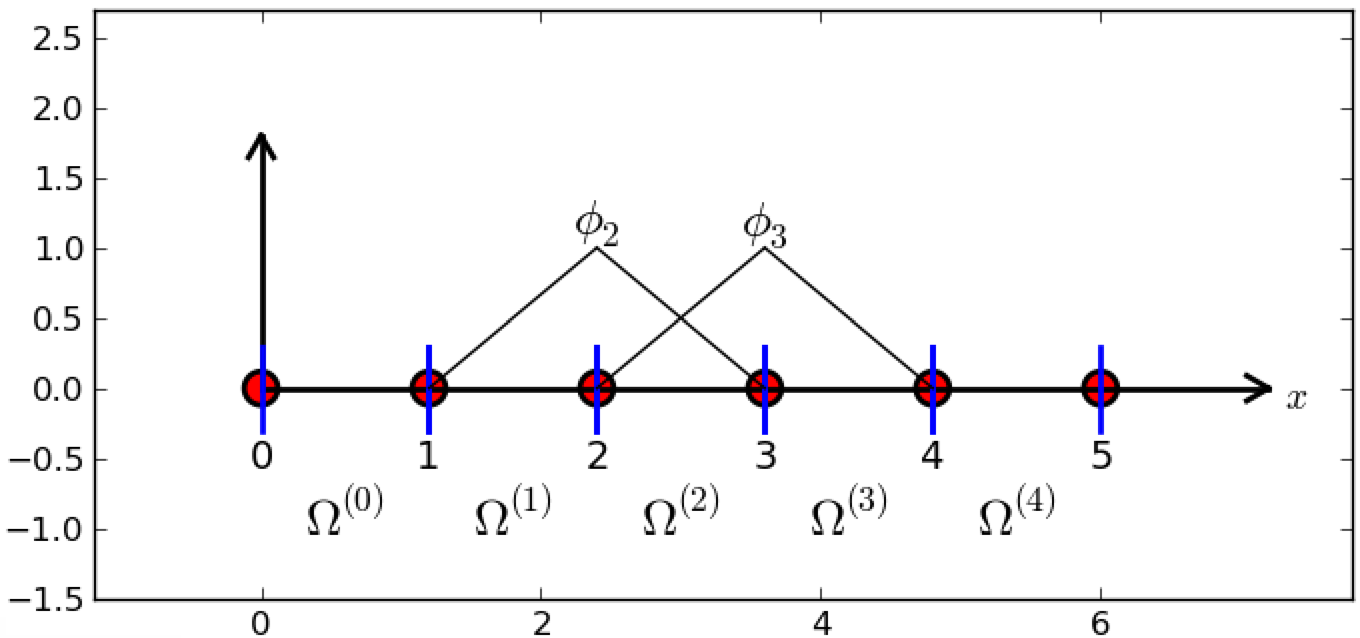
$$\left(\frac{-q''_{ws} R}{r} - h_{int}(T_m - T_l)\right) = \left(1 + \left(\frac{dr_i}{dz}\right)^2\right) \frac{\partial r_i}{\partial t} (\rho \Delta H)$$

3. Physics in MOOSE/SAM Recap

```
Real
FluidPressureAreaTimeDerivative::computeQpOffDiagJacobian(unsigned int jvar)
{
  Real area = M_PI*_radiusi[_qp]*_radiusi[_qp];
  Real area_dot = M_PI*2*_radiusi[_qp]*_radiusi_dot[_qp];

  if (jvar == _temperature_var_number){
    Real drho_dT = _eos.drho_dT(_u[_qp],_temperature[_qp]);
    return drho_dT*_phi[_j][_qp]*_test[_i][_qp]*area_dot/area;
  }
  if (jvar == _radiusi_var_number){
    Real first_part = _phi[_j][_qp]*_d_radiusidot_du[_qp]/_radiusi[_qp];
    Real second_part = -_radiusi_dot[_qp]*_phi[_j][_qp]/(_radiusi[_qp]*_radiusi[_qp]);
    return (first_part + second_part)*2*_rho[_qp]*_test[_i][_qp];
  }
  else  return 0.;
}
```

FEM Matrix equation to solve a PDE

- Take the variational form and discretize
- (V_h) $(u', v') = (f, v) \quad \forall v \in V_h \quad Find\ u \in V_h$
- $V_h = span\{\phi_i\}$ so all v are linear combinations of ϕ_i
- $(u', \phi_i{}') = (f, \phi_i) \quad \forall \phi_i \quad i = 1 \dots M$
- $\left(\sum c_j \phi_j{}', \phi_i{}'\right) = (f, \phi_i) \quad \forall \phi_i \quad i = 1 \dots M$
- $\sum_{j=1}^{j=M} (\phi_j{}', \phi_i{}') c_j = (f, \phi_i) \quad \forall \phi_i \quad i = 1 \dots M$



Quick Example – a time derivative term

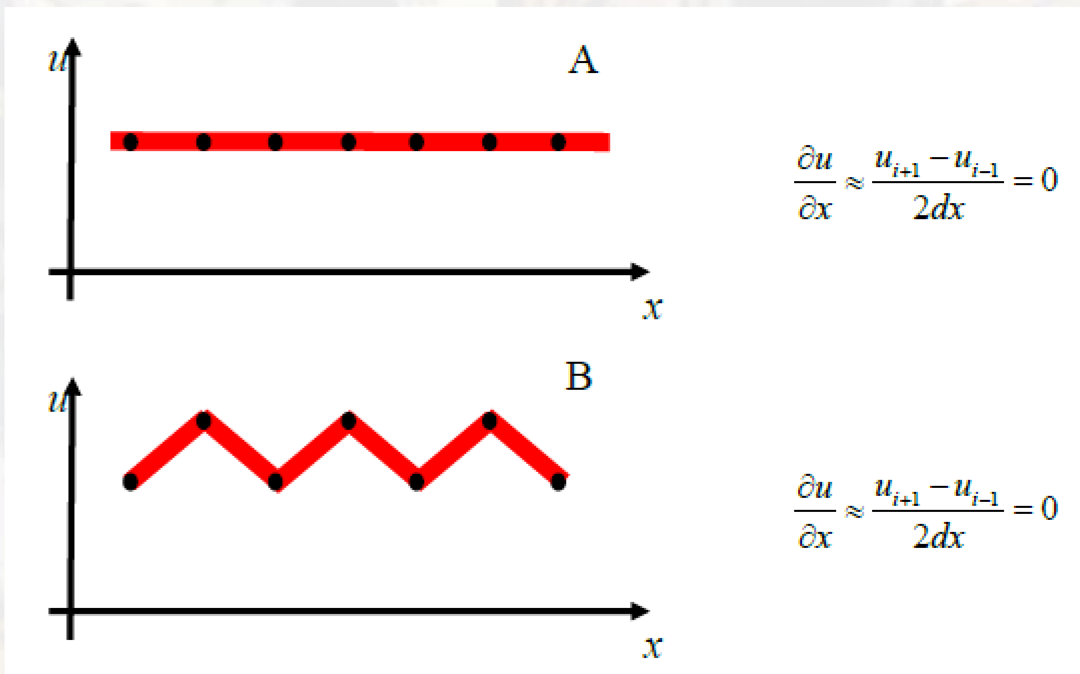
- $\frac{\partial \rho_l}{\partial t} + \frac{\partial (\rho u)_l}{dz} + \frac{\rho_l}{A_l}\frac{\partial A_l}{\partial t} + \frac{(\rho u)_l}{A_l}\frac{\partial A_l}{dz} = -\frac{\sigma_A}{A}$
- $R = \frac{\rho_l}{A_l}\frac{\partial A_l}{\partial t}\psi = \frac{2\rho_l}{r_i}\frac{\partial r_i}{\partial t}\psi$ where $\frac{\partial A_l}{\partial t} = 2\pi r_i \frac{\partial r_i}{\partial t}$
- $J(T) = \frac{1}{A_l}\frac{\partial \rho}{\partial T}\varphi_j \frac{\partial A_l}{\partial t}\psi$
- $J(r) = \frac{2\rho_l \psi}{r_i^2}\left(r\frac{\partial \dot{r}}{dr}\varphi_j - \frac{\partial r_i}{dt}\varphi_j\right)$
- $= 2\pi r_i \frac{\rho_l}{A_l}\psi\left(\frac{\partial \dot{r}}{dr}\varphi_j - \frac{\partial A_l}{dt}\frac{\varphi_j}{2A_l}\right)$

Quick Example – a time derivative term

- $$\frac{\partial \rho_l}{\partial t} + \frac{\partial (\rho u)_l}{dz} + \frac{\rho_l}{A_l}\frac{\partial A_l}{\partial t} + \frac{(\rho u)_l}{A_l}\frac{\partial A_l}{dz} = -\frac{\sigma_A}{A}$$
- $$R = \frac{\rho_l}{A_l}\frac{\partial A_l}{\partial t}\psi = \frac{2\rho_l}{r_i}\frac{\partial r_i}{\partial t}\psi \quad \text{where} \quad \frac{\partial A_l}{\partial t} = 2\pi r_i \frac{\partial r_i}{\partial t}$$
- $$J(T) = \frac{1}{A_l}\frac{\partial \rho}{\partial T}\varphi_j \frac{\partial A_l}{\partial t}$$
- $$J(r) = \frac{2\rho_l \psi}{r_i^2}\left(r\frac{\partial \dot{r}}{dr}\varphi\right.$$
- $$= 2\pi r_i \frac{\rho_l}{A_l}\psi\left(\frac{\partial \dot{r}}{dr}\varphi_j\right.$$

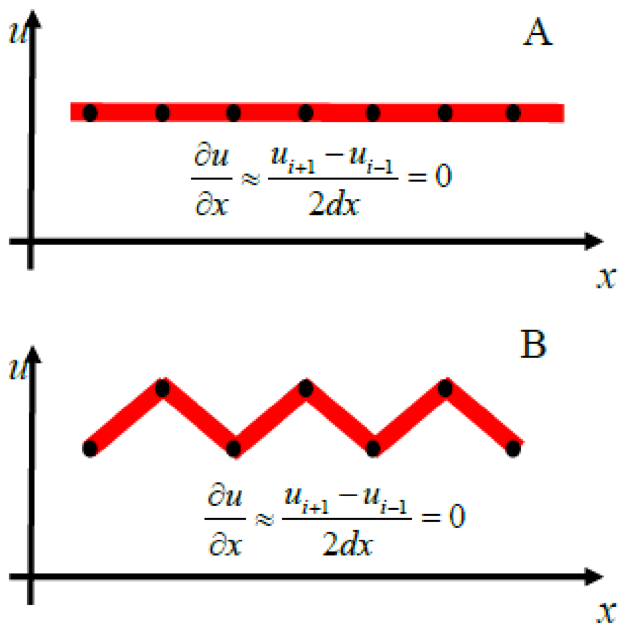
```
Real
FluidPressureAreaTimeDerivative::computeQpResidual()
{
  Real area = M_PI*_radiusi[_qp]*_radiusi[_qp];
  Real area_dot = M_PI*2*_radiusi[_qp]*_radiusi_dot[_qp];

  return  _rho[_qp] * _test[_i][_qp] * area_dot/area;
}

Real
FluidPressureAreaTimeDerivative::computeQpJacobian()
{
  return 0;
}

Real
FluidPressureAreaTimeDerivative::computeQpOffDiagJacobian(unsigned int jvar)
{
  Real area = M_PI*_radiusi[_qp]*_radiusi[_qp];
  Real area_dot = M_PI*2*_radiusi[_qp]*_radiusi_dot[_qp];

  if (jvar == _temperature_var_number){
    Real drho_dT = _eos.drho_dT(_u[_qp],_temperature[_qp]);
    return drho_dT*_phi[_j][_qp]*_test[_i][_qp]*area_dot/area;
  }
  if (jvar == _radiusi_var_number){
    Real first_part = _phi[_j][_qp]*_d_radiusidot_du[_qp]/_radiusi[_qp];
    Real second_part = -_radiusi_dot[_qp]*_phi[_j][_qp]/(_radiusi[_qp]*_radiusi[_qp]);
    return (first_part + second_part)*2*_rho[_qp]*_test[_i][_qp];
  }
  else  return 0.;
}
```

4. Issues with Numerics

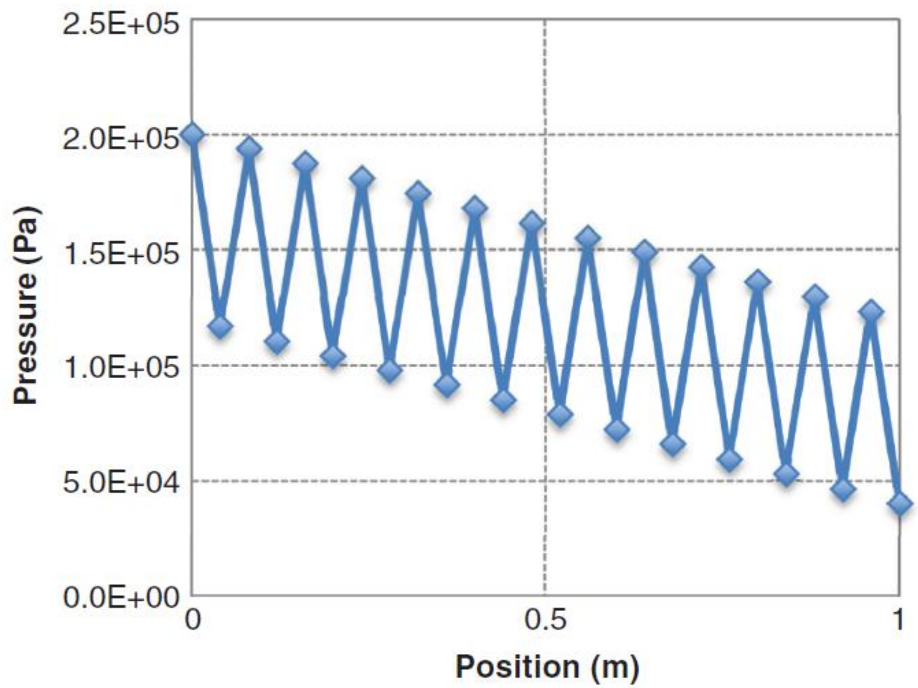


Standard NS Equations: Poor Numerics

- Can have non-physical numerical oscillation
- Notice, for example, a common issue with central differences
  - With other CFD methods, might be remedied with artificial numerical diffusion – can cost an order of accuracy
- Poor conditioning – lack of Pressure in Pressure eq: $$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{dz} = 0$$

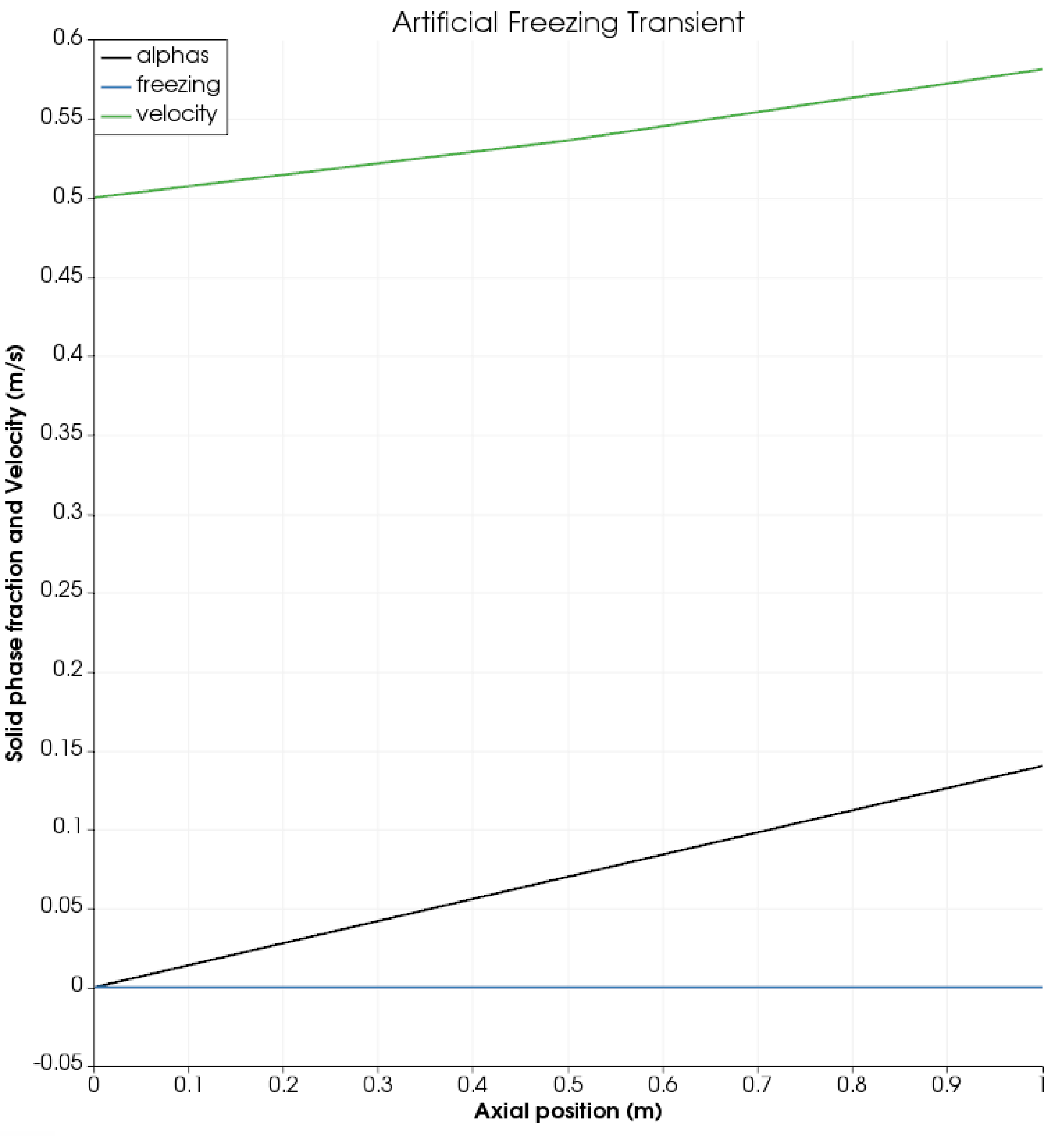


C. Rutland, "Stability Part 2; Advanced Numerical Analysis"

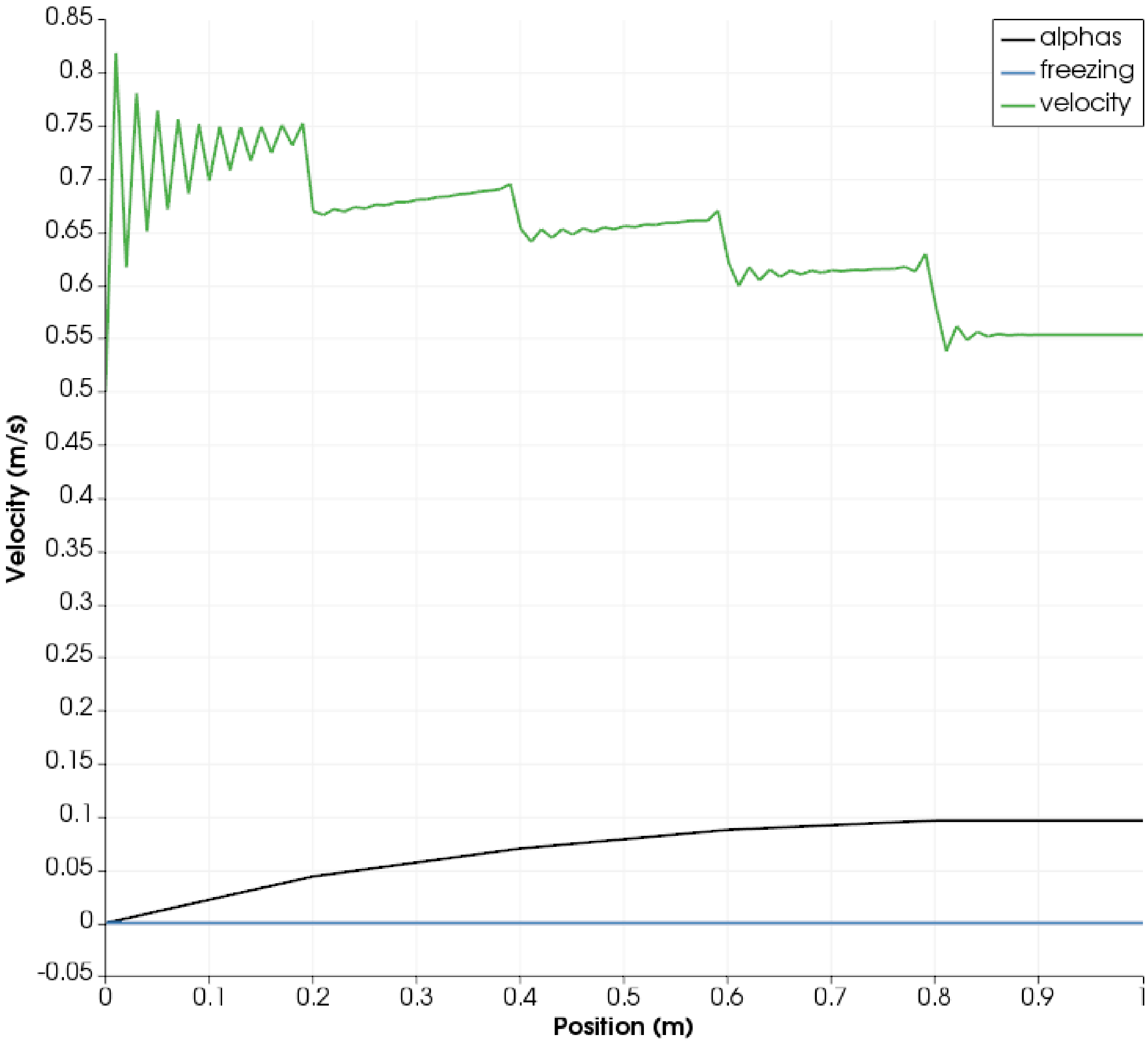


R. Hu, "An Advanced One-Dimensional Finite Element Model…"

Some cases will function, others will not

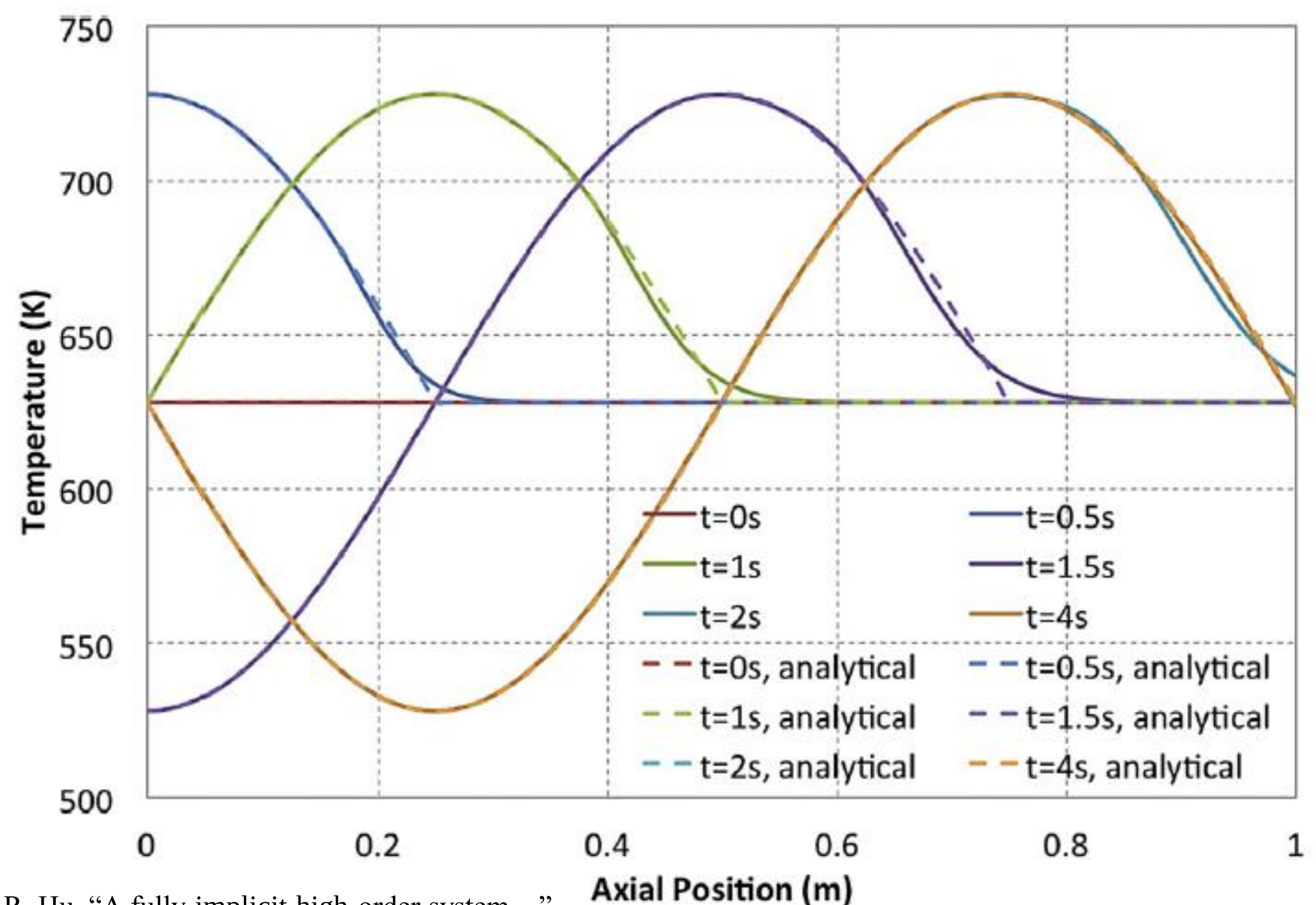


Artificial Freezing Transient – Garbage out



DEPARTMENT OF Engineering Physics
UNIVERSITY OF WISCONSIN–MADISON

4. Stabilization



R. Hu, “A fully-implicit high-order system…”

PSPG/SUPG Formulation: Mass Equation

- Pressure-Stabilized / Streamline-Upwind Petrov-Galerkin, adapted
- *momentum* $\frac{\partial(\alpha\rho u)_l}{\partial t} + \frac{\partial(\alpha\rho u u)_l}{\partial z} + \frac{\partial(\alpha p)_l}{\partial z} + \rho g + \frac{f}{D}\frac{\rho u|u|}{2} = 0$
- *and continuity* $\left(\frac{\partial(\alpha\rho)_l}{\partial t} + \frac{\partial(\alpha\rho u)_l}{\partial z} = -\frac{\sigma_A}{A}\right) * u \quad \Rightarrow$
- $\left((\alpha\rho)_l\frac{\partial u_l}{\partial t} + (\alpha\rho u)_l\frac{\partial u_l}{\partial z} + \frac{\partial(\alpha p)_l}{\partial z} + \rho g + \frac{f}{D}\frac{\rho u|u|}{2} - \frac{u\sigma_A}{A}, \tau_{PSPG}\frac{\partial\psi}{\partial z}\right) + \left(\frac{\partial(\alpha\rho)_l}{\partial t} + \frac{\partial(\alpha\rho u)_l}{\partial z} + \frac{\sigma_A}{A}, \psi\right) = 0$ *new mass equation*
- $\tau_{PSPG} = \left[\left(\frac{2}{\Delta t}\right)^2 + \left(\frac{2|u|}{h}\right)^2 + \left(\frac{4\nu}{h^2}\right)^2\right]^{-\frac{1}{2}}$ $\qquad \tau_{SUPG} = u * \tau_{PSPG}$

PSPG/SUPG Formulation: Momentum/Velocity

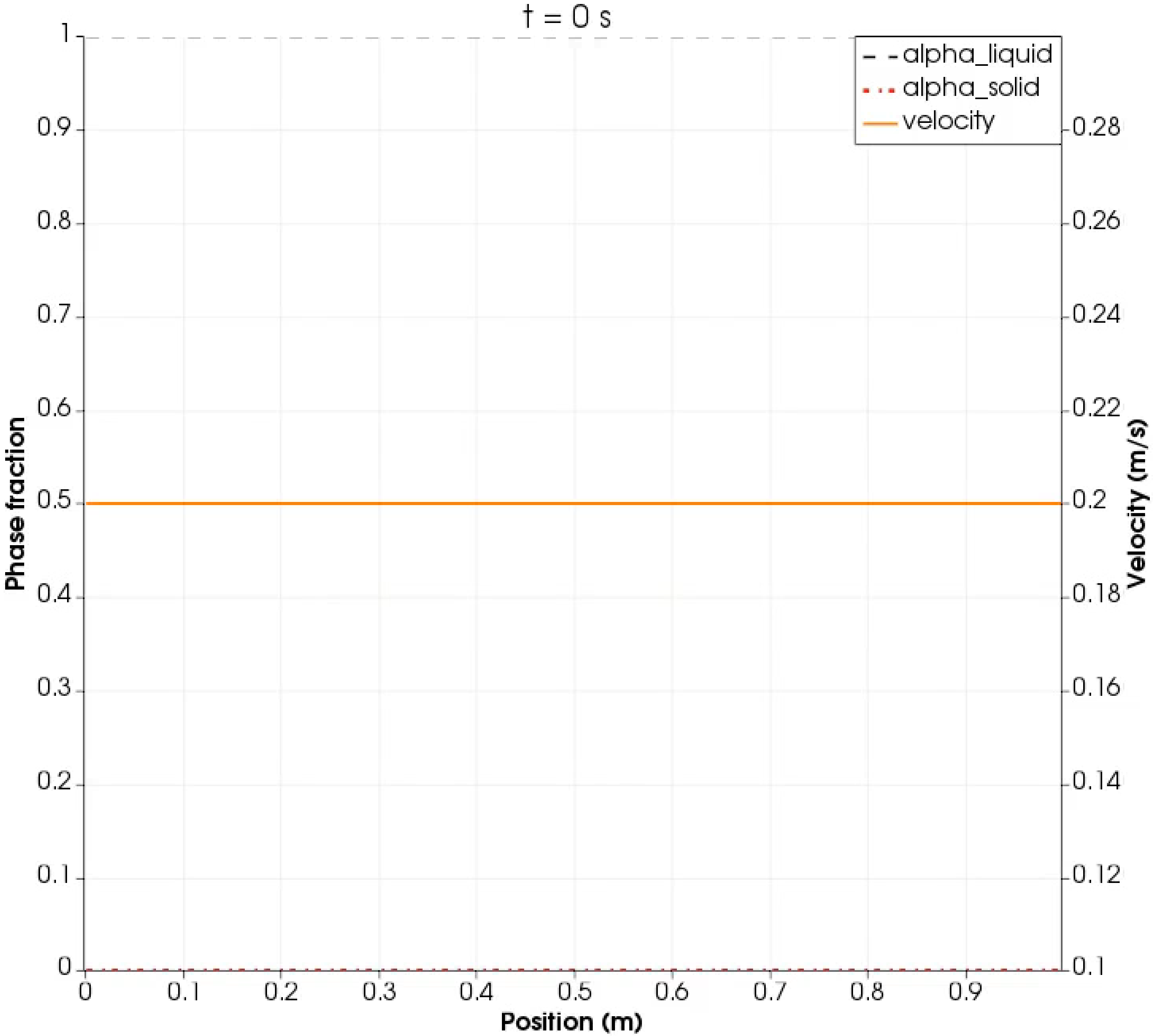
- $\left((\alpha\rho)_l \frac{\partial u_l}{\partial t} + (\alpha\rho u)_l \frac{\partial u_l}{\partial z} + \frac{\partial(\alpha p)_l}{\partial z} + \rho g + \frac{f}{D}\frac{\rho u|u|}{2} - \frac{u\sigma_A}{A}, \tau_{SUPG}\frac{\partial\psi}{\partial z}\right) + \left(\frac{\partial(\alpha\rho u)_l}{\partial t} + \frac{\partial(\alpha\rho u u + \alpha p)_l}{\partial z} + \rho g + \frac{f}{D}\frac{\rho u|u|}{2}, \psi\right) = 0 \qquad \textit{new velocity equation}$

- Consider the gradient term
- $\int_\Omega \frac{\partial(\alpha\rho u u + \alpha p)_l}{\partial z}\psi = \int_\Omega \frac{\partial(\alpha\rho u u\psi + \alpha p\psi)_l}{\partial z} - \int_\Omega \frac{\partial\psi}{\partial z}(\alpha\rho u u + \alpha p)_l$

- Now apply divergence theorem to the first term
- $= \underbrace{\int_{\partial\Omega}(\alpha\rho u u + \alpha p)_l \psi \cdot \hat{n}\, dS}_{Boundary} - \underbrace{\int_\Omega \frac{\partial\psi}{\partial z}(\alpha\rho u u + \alpha p)_l}_{Residual}$

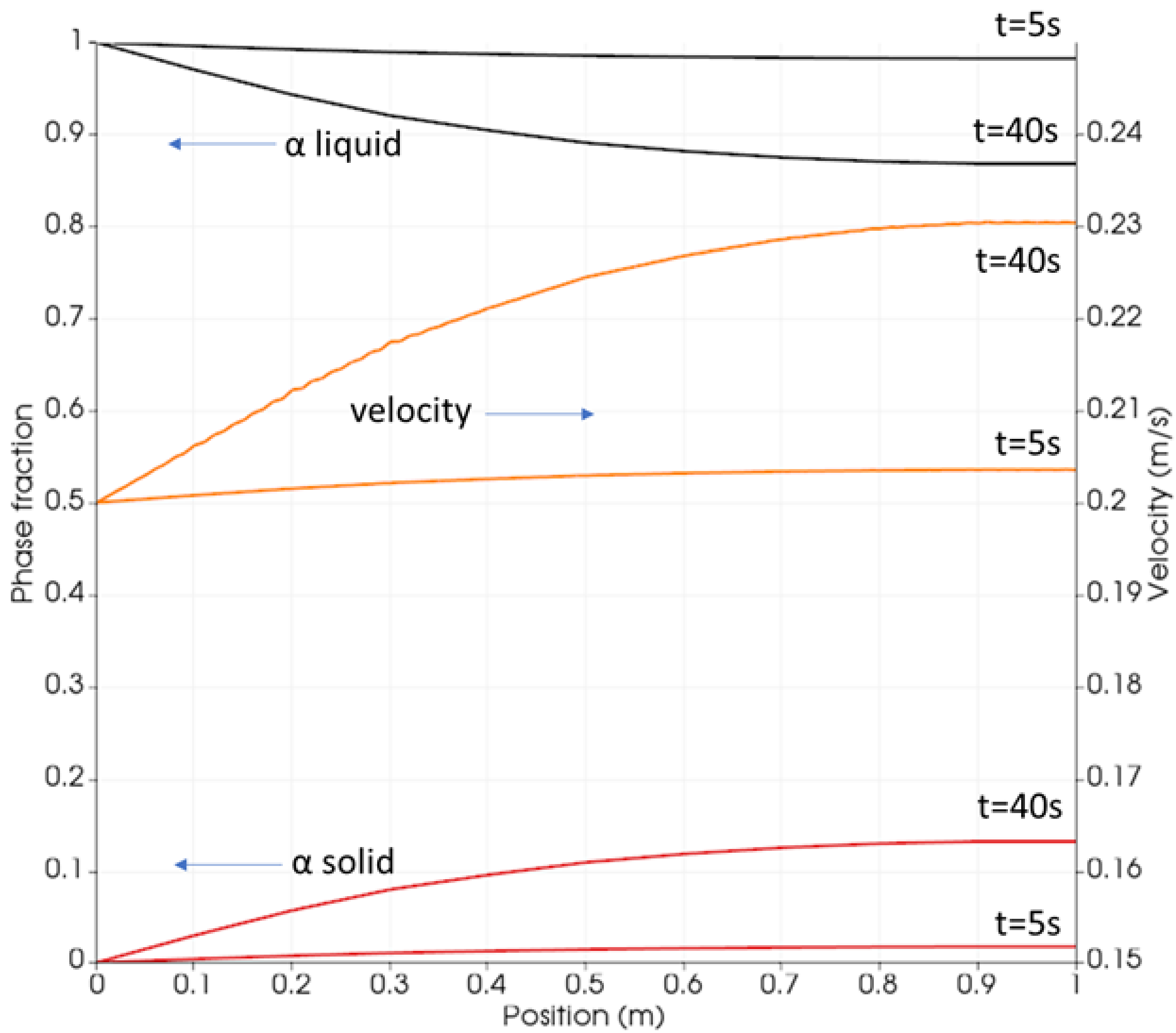
- $\Rightarrow R = \left(\rho g + \frac{f}{D}\frac{\rho u|u|}{2}\right)_l \psi - \frac{\partial\psi}{\partial z}(\alpha\rho u u + \alpha p)_l +$
$$\tau_{SUPG}\frac{\partial\psi}{\partial z} * \left(\alpha\rho\frac{\partial u}{\partial t} + \alpha\rho u\frac{\partial u}{\partial z} + \frac{\partial(\alpha p)}{\partial z} + \rho g + \frac{f}{D}\frac{\rho u|u|}{2} - \frac{u\sigma_A}{A}\right)_l$$

Bas



t = 0 s
alpha_liquid
alpha_solid
velocity
Phase fraction
Velocity (m/s)
Position (m)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0.1
0.12
0.14
0.16
0.18
0.2
0.22
0.24
0.26
0.28

Basic Test: “3 Equation” Example



PSPG/SUPG Formulation: Energy

- Similarly for the Energy/Temperature equation:

- $\frac{\partial(\alpha\rho H)_l}{\partial t} + \frac{\partial(\alpha\rho u H)_l}{dz} = -\frac{\sigma_A H}{A} + q'''$

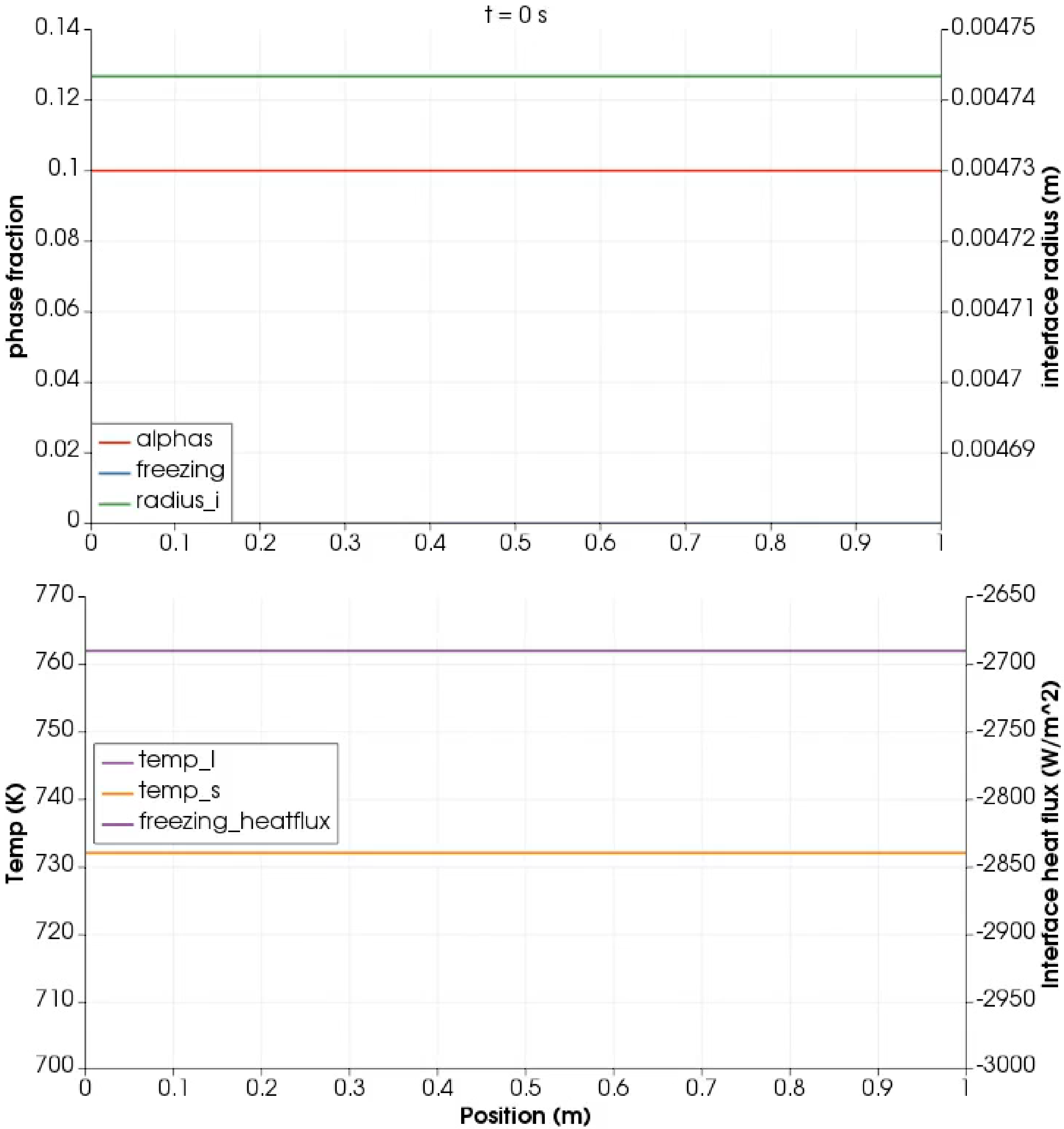
- $\Rightarrow \left(\alpha\rho c_p \frac{\partial T}{\partial t} + \alpha\rho c_p u \frac{\partial T}{\partial z} - q''', \tau_{SUPG} \frac{\partial\psi}{\partial z}\right) + \left(\frac{\partial(\alpha\rho H)_l}{\partial t} + \frac{\partial(\alpha\rho u H)_l}{\partial z} - q''' + \frac{\sigma_A H}{A}, \psi\right) = 0$

- Consider the gradient term
- $\int_\Omega \frac{\partial(\alpha\rho u H)_l}{\partial z}\psi = \int_\Omega \frac{\partial(\alpha\rho u H\psi)_l}{\partial z} - \int_\Omega \frac{\partial\psi}{\partial z}(\alpha\rho u H)_l$

- Now apply divergence theorem to the first term
- $= \underbrace{\int_{\partial\Omega}(\alpha\rho u H)_l \psi \cdot \hat{n}\, dS}_{Boundary} - \underbrace{\int_\Omega \frac{\partial\psi}{\partial z}(\alpha\rho u H)_l}_{Residual}$

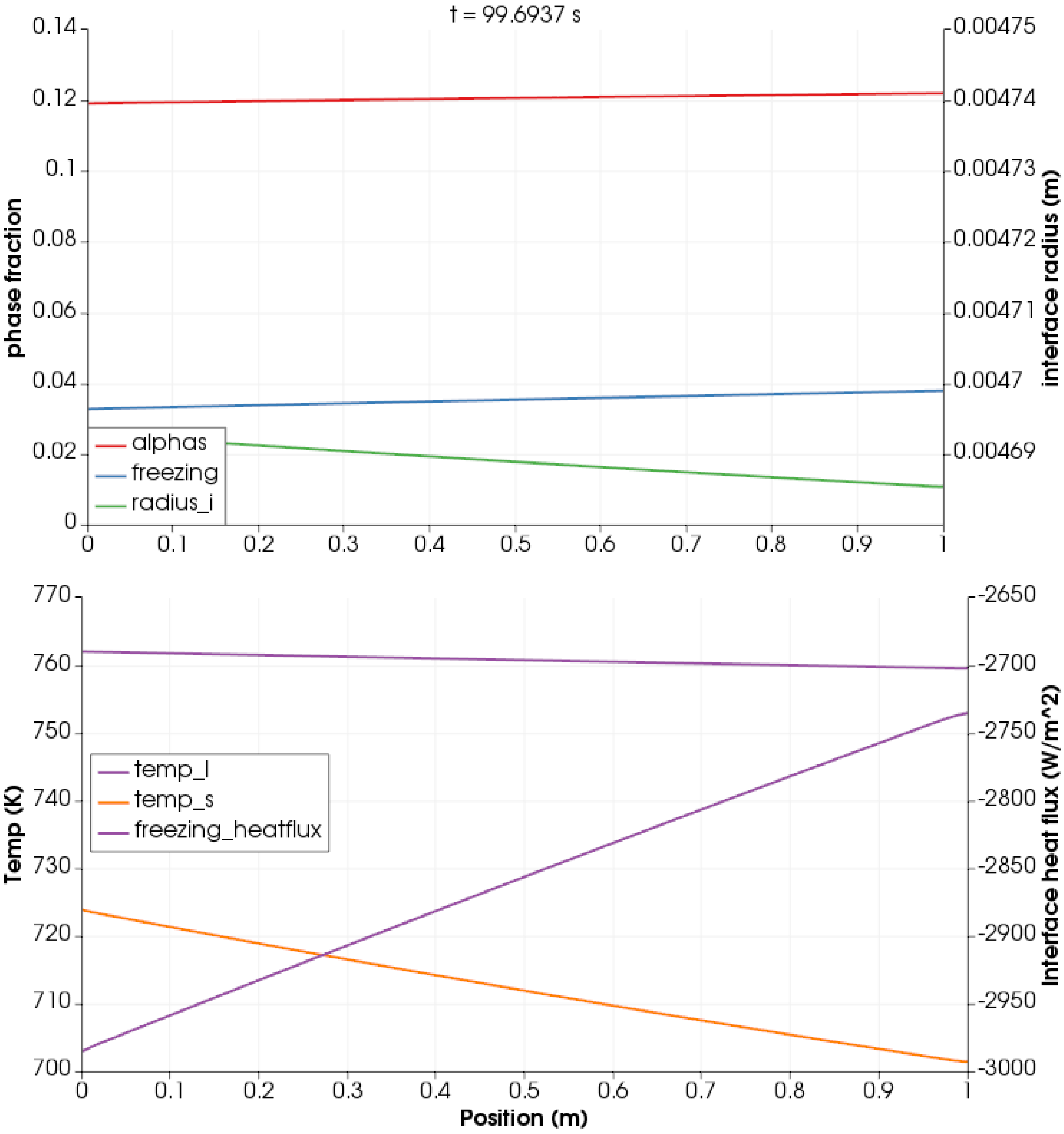
```
Real OneDFluidTemperatureAlpha::computeQpJacobian()
{
  Real Cp = _eos.c_p(_pressure[_qp],_u[_qp]);
  Real enth = _eos.enthalpy_from_T(_u[_qp]);
  Real alphal = (1.0 - _alphas[_qp]);
  Real tao = computeTaoSUPG();
  Real drho_dT = _eos.drho_dT(_pressure[_qp],_u[_qp]);
  Real dCpdT = _eos.dCp_dT(_pressure[_qp],_u[_qp]);

  Real psi = _test[_i][_qp];
  Real dpsidx = _grad_test[_i][_qp](0);
  Real psi_supg = tao*_grad_test[_i][_qp](0);

  //Real diffusion_part = 0;
  //if (_fcm)  diffusion_part = k * _grad_phi[_j][_qp](0) * dpsidx;

  Real convect1_part = -alphal *  _rho[_qp] * _velocity[_qp] * Cp * _phi[_j][_qp] * dpsidx;
  Real convect2_part = -alphal * drho_dT * _phi[_j][_qp] * _velocity[_qp] * enth * dpsidx;
  Real source_part = (_freezing[_qp] / _Ax) * Cp * _phi[_j][_qp] * _test[_i][_qp];
  Real normal_part = convect1_part + convect2_part + source_part;

  Real tran1_supg = alphal * drho_dT * _phi[_j][_qp] * Cp * _u_dot[_qp];
  Real tran2_supg = alphal * _rho[_qp] * dCpdT * _phi[_j][_qp] * _u_dot[_qp];
  Real tran3_supg = alphal * _rho[_qp] * Cp * _du_dot_du[_qp] * _phi[_j][_qp];
  Real conv1_supg = alphal * drho_dT * _phi[_j][_qp] * Cp * _velocity[_qp] * _grad_u[_qp](0);
  Real conv2_supg = alphal * _rho[_qp] * dCpdT * _phi[_j][_qp] *  _velocity[_qp] * _grad_u[_qp](0);
  Real conv3_supg = alphal * _rho[_qp] * Cp * _velocity[_qp] * _grad_phi[_j][_qp](0);
  Real supg_part = psi_supg*(tran1_supg+tran2_supg+tran3_supg + conv1_supg+conv2_supg+conv3_supg);

  return (normal_part + supg_part);
}
```

Basic



t = 0 s
phase fraction
interface radius (m)
alphas
freezing
radius_i
Temp (K)
Interface heat flux (W/m^2)
temp_l
temp_s
freezing_heatflux
Position (m)

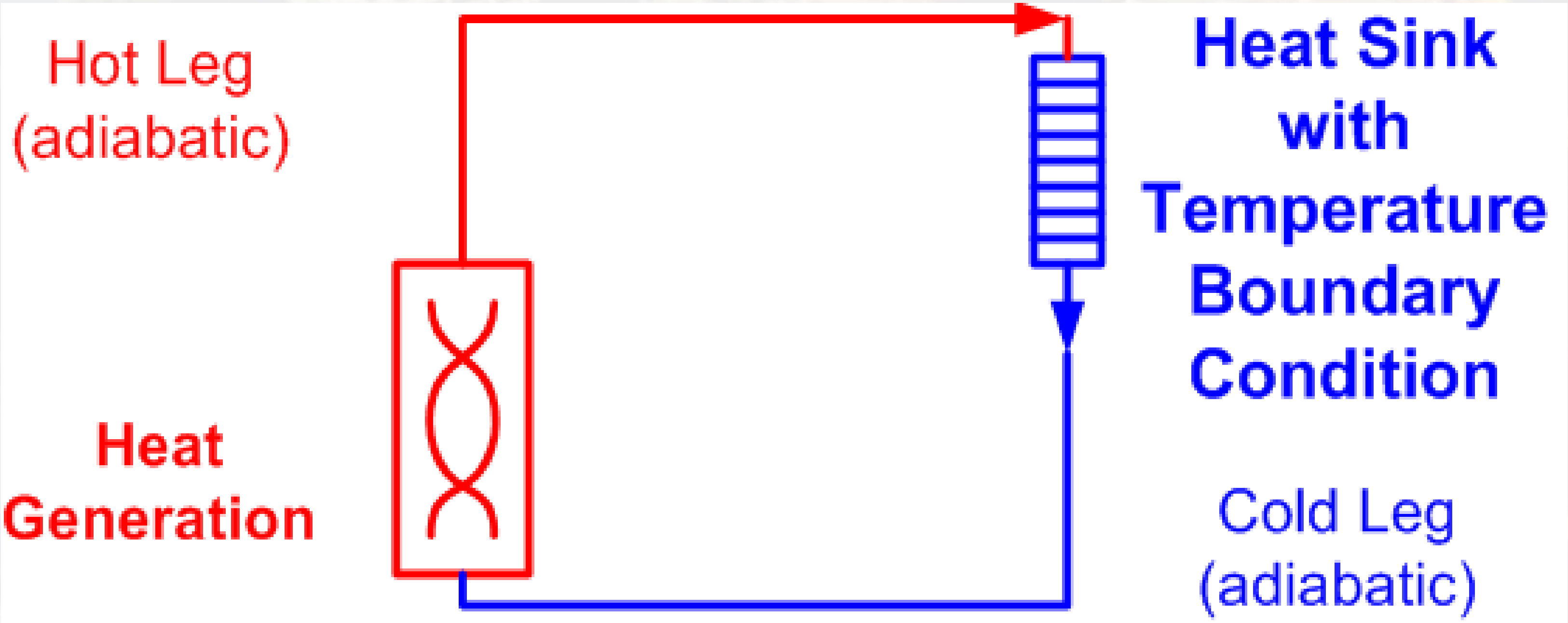
Basic



t = 99.6937 s
phase fraction
interface radius (m)
alphas
freezing
radius_i
Temp (K)
Interface heat flux (W/m^2)
temp_l
temp_s
freezing_heatflux
Position (m)

DEPARTMENT OF Engineering Physics
UNIVERSITY OF WISCONSIN–MADISON

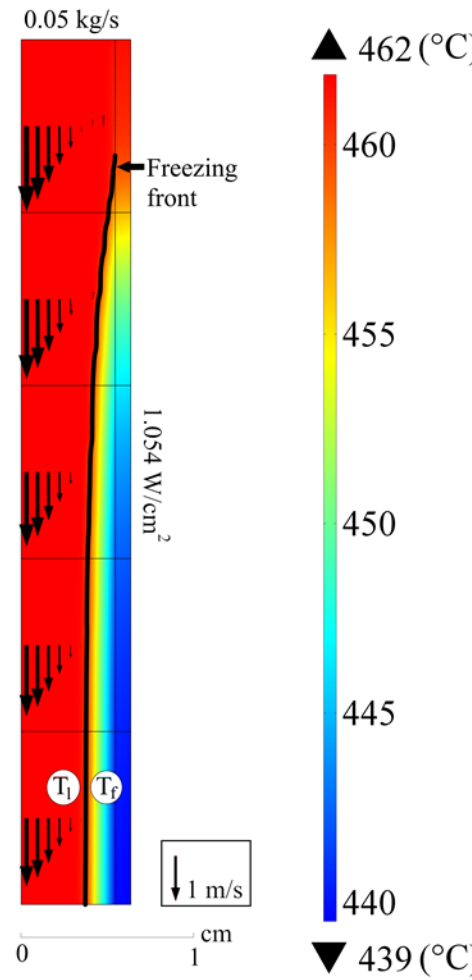
6. Path Forward



Next steps

- Kernels -> Components
- Components -> Tests
- Components -> Benchmarks
- CFD – improve freezing closure
- Add overcooling pipe component to natural circulation loop

Pipe Freezing Simulation

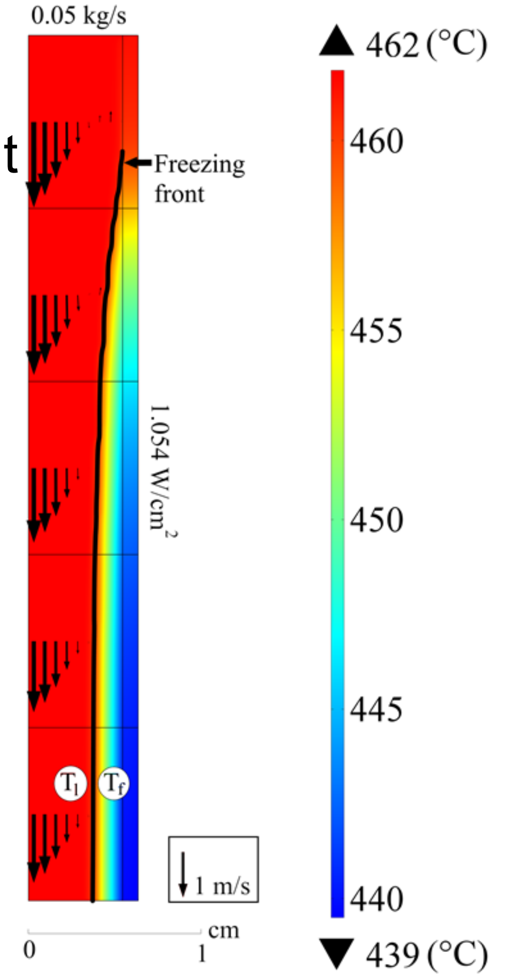


Physics: Radiative heat transfer, bouyancy force, axisymmertic

Challenges

- Representing a PDE is relatively straightforward
- Making a robust, flexible freezing component isn't
  - Account for proper **freezing initiation** time considering the bulk fluid will be above T_m
  - Predict freeze vs melt, and the code should handle the **frozen fraction going to zero**
  - Must be implemented in a way that **avoids inconsistencies** – solid temperature shouldn't go above freezing, phase fractions should always be between 0 and 1, etc.

Pipe Freezing Simulation



Physics: Radiative heat transfer, bouyancy force, axisymmertic

References

- C. Rutland, “Stability Part 2; Advanced Numerical Analysis”, ME573 – Computational Fluid Dynamics, Fall 2016, University of Wisconsin-Madison.
- R. Hu, “An Advanced One-Dimensional Finite Element Model for Incompressible Thermally Expandable Flow”, Argonne National Laboratory, in *Nuclear Technology Vol. 190*, June 2015.
- R. Hu, “A fully-implicit high-order system thermal-hydraulics model for advanced non-LWR safety analyses”, Argonne National Laboratory, in *Annals of Nuclear Energy 101 (2017) 174-181*.

Acknowledgements

The work is supported under NEUP Grant DE-NE0008545 (Project 16-10647) Experimental and Modeling Investigation of Overcooling Transients that include Freezing, in Fluoride-Salt Cooled High-Temperature Reactors (FHRs). This material is based upon work supported under an Integrated University Program Graduate Fellowship.

Acknowledgements to Dr. Rui Hu and Argonne National Laboratory for continued support and development of the SAM code

1D FREEZING MODEL APPROACH

A FREEZING MODEL SUITABLE FOR USE IN SAM SHOULD BE DEVELOPED

The model can be similar in form to what SAM currently uses

- For reference, the conservative governing equations from SAM are as follows[5]

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{dz} = 0$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u u + p)}{dz} = -\rho g - \frac{f}{D}\frac{\rho u|u|}{2}$$

$$\frac{\partial(\rho H)}{\partial t} + \frac{\partial(\rho u H)}{dz} = \frac{q_s'' P_h}{A_c}$$

- For a freeze-capable model, start with the area-averaged generic balance with storage, convection, diffusion, body, and surface terms[6]

$$\frac{\partial(A\rho\psi)}{\partial t} + \frac{\partial(A\rho\psi u \cdot \hat{n}_z)}{dz} + \frac{\partial(AJ \cdot \hat{n}_z)}{dz} - A\rho\phi = -\int_c J \cdot \hat{n}\, dS$$

THE MASS EQUATION

We'll write all the balance equations with varying area

- The mass equation simplifies in a straightforward manner. At this stage also define a source term that is essentially a linear mass transfer rate, and write the liquid and solid equations

$$\frac{\partial (A\rho)_l}{\partial t} + \frac{\partial (A\rho u)_l}{dz} = -\sigma_A \quad \left[\frac{kg}{m \cdot s}\right]$$

$$\frac{\partial (A\rho)_s}{\partial t} = \sigma_A$$

Argonne NATIONAL LABORATORY

THE MOMENTUM EQUATION

Momentum becomes essentially the same equation with varying area

- For the momentum equation, the general expression is

$$\frac{\partial(A\rho u)}{\partial t} + \frac{\partial(A\rho uu)}{dz} + \frac{\partial(Ap\hat{n}_z - A\tau \cdot \hat{n}_z)}{dz} - A\rho g = \int_c -pI \cdot \hat{n} + \tau \cdot \hat{n} \, dS$$

- Taking the scalar product with $\hat{n}_z$, using $\frac{\partial A}{\partial z} = -\int_c \hat{n} \cdot \hat{n}_z \, dS$ (geometric relation describing the cross sectional area change as an observer traverses the z-axis of a cone), and rearranging

$$\frac{\partial(A\rho u)}{\partial t} + \frac{\partial(A\rho uu + p)}{dz} - \frac{\partial(A(\tau \cdot \hat{n}_z) \cdot \hat{n}_z)}{dz} = A\rho g + p\frac{\partial A}{\partial z} + \int_c \tau \cdot \hat{n} \, dS$$

THE MOMENTUM EQUATION

Momentum becomes essentially the same equation with varying area

- Neglect dissipation, (fine for small Ec, for expected conditions with salt ~10^{-7}), add source term, and use a friction factor term in place of the right side integral to obtain

$$\frac{\partial(A\rho u)}{\partial t} + \frac{\partial(A\rho uu + p)}{dz} = A\rho g + p\frac{\partial A}{\partial z} + \frac{Af}{D}\frac{\rho u|u|}{2} - \sigma_A u_{int}$$

- Assuming the interfacial velocity is zero, and compression/acceleration term negligible

$$\frac{\partial(A\rho u)}{\partial t} + \frac{\partial(A\rho uu + p)}{dz} = A\rho g + \frac{Af}{D}\frac{\rho u|u|}{2}$$

THE ENERGY EQUATION

Heat removal with latent heat transfer remains the most irresolute aspect of this model

- For the energy equation, the general expression is

$$\frac{\partial(A\rho E)}{\partial t} + \frac{\partial(A\rho E u \cdot \hat{n}_z)}{dz} + \frac{\partial((Aq + ApI \cdot u - A\tau \cdot u) \cdot \hat{n}_z)}{dz} - A\rho\left(g \cdot u + \frac{r}{\rho}\right)$$
$$= -\int_c (q + pI \cdot u - \tau \cdot u) \cdot \hat{n}_z \, dS$$

- Neglect shear stress and dissipation (tau terms on both sides), and body terms on left side

$$\frac{\partial(A\rho E)}{\partial t} + \frac{\partial(A\rho E u)}{dz} + \frac{\partial(Aq + Apu)}{dz} = -\int_c pu \cdot \hat{n}_z \, dS - \int_c q \cdot \hat{n}_z \, dS$$

THE ENERGY EQUATION

Heat removal with latent heat transfer remains the most irresolute aspect of this model

- Neglect axial conduction and combine terms

$$\frac{\partial(A\rho E)}{\partial t} + \frac{\partial(Au(\rho E + p))}{dz} = -\int_c pu \cdot \hat{n}_z \, dS - \int_c q \cdot \hat{n}_z \, dS$$

- Using $\frac{\partial A}{\partial t} = \int_c u_{wall} \cdot \hat{n} \;\, dS$ (geometric relation describing the cross sectional area change as the perimeter of a cone narrows), and rewriting the heat term

$$\frac{\partial(A\rho E)}{\partial t} + \frac{\partial(Au(\rho E + p))}{dz} = -p\frac{\partial A}{\partial t} + q''_{ls} P_{ls}$$

THE ENERGY EQUATION

Heat removal with latent heat transfer remains the most irresolute aspect of this model

- Add a source term related to mass transfer, and neglect the compression/acceleration term

$$\frac{\partial(A\rho E)}{\partial t} + \frac{\partial(Au(\rho E + p))}{dz} = -\sigma_A E + q''_{ls} P_{ls}$$

- While the second term on the left represents enthalpy exactly, for this incompressible model making the further assumption $\Delta H = \Delta E$ provides

$$\frac{\partial(A\rho H)}{\partial t} + \frac{\partial(A\rho u H)}{dz} = -\sigma_A H + q''_{ls} P_{ls}$$

THE ENERGY EQUATION

Heat removal with latent heat transfer remains the most irresolute aspect of this model

- The heat transfer term could be changed to a conditional version based on existence of a frozen layer of coolant

$$\frac{\partial(A\rho H)_l}{\partial t} + \frac{\partial(A\rho uH)_l}{dz} = -\sigma_A H + q'_{wl}\left(T_l > T_f\right) + q'_{int}\left(T_l \leq T_f\right)$$

- Where the terms for heat transfer out of the component and heat transfer from the liquid part to the solid part are

$$q'_{wl} = h(T_\infty - T_l)P_H \qquad \text{(to be updated to radiative)}$$
$$q'_{int} = h_{int}(T_m - T_l)P_{int}\ \{+\sigma_A(\Delta H_f)\}$$

THE ENERGY EQUATION

Heat removal with latent heat transfer remains the most irresolute aspect of this model

- The solid energy equation is

$$\frac{\partial (A\rho H)_s}{\partial t} = \sigma_A H_s + q'_{ws} - (q'_{int} + \{\sigma_A(\Delta H_f)\})$$

$$q'_{ws} = h(T_\infty - T_s)P_H$$ (to be updated to radiative)

Argonne NATIONAL LABORATORY

THE FREEZING INITIATION TEMPERATURE

Freezing will begin at a bulk temperature higher than the melting point of the salt

- With ΔT_s is the difference between the bulk temperature and the surface temperature

$$T_f = T_m + \Delta T_s$$

- From the analytical profile shown earlier

$$T_{min} = T_b + \frac{11}{24}\frac{q_w'' R}{k}$$

$$\Delta T_s = -\frac{11}{24}\frac{q_w'' R}{k}$$

Argonne NATIONAL LABORATORY

CLOSURE NEEDED FOR THE FREEZING RATE

Physically this depends on the temperature gradients at the interface

- Semi-empirical and/or CFD-based approach may be necessary; for now consider simplified analytical approach

- With a heat balance at the interface, the freeze front is stationary when $q_l'' = q_s''$. The interface propagates when these are not balanced, through latent heat storage or release, where $\hat{n}$ is normal to the interface

$$(q_l'' - q_s'') = \frac{\partial \hat{n}}{\partial t}(\rho \Delta H)$$

$$\left(k_s \frac{\partial T_s}{\partial \hat{n}} - k_l \frac{\partial T_l}{\partial \hat{n}}\right) = \frac{\partial \hat{n}}{\partial t}(\rho \Delta H)$$

CLOSURE NEEDED FOR THE FREEZING RATE

Physically this depends on the temperature gradients at the interface

- For an interface that changes gradually in the z-direction, this form approximates the expression for the r-direction, but to express it more rigorously, use the geometric relations:

$$\frac{\partial \hat{n}}{\partial t} = \frac{\partial \hat{n}}{\partial r}\frac{\partial r}{\partial t} \qquad \frac{\partial T}{\partial \hat{n}} = \frac{\partial T}{\partial r}\frac{\partial r}{\partial \hat{n}} \qquad \left(\frac{\partial r}{\partial z}\right)^2 + 1 = \left(\frac{\partial \hat{n}}{dr}\right)^2$$

- Therefore

$$\frac{\partial r}{\partial \hat{n}}\left(k_s \frac{\partial T_s}{\partial r} - k_l \frac{\partial T_l}{\partial r}\right) = \frac{\partial \hat{n}}{\partial r}\frac{\partial r}{\partial t}(\rho \Delta H)$$

$$\left(k_s \frac{\partial T_s}{\partial r_i} - k_l \frac{\partial T_l}{\partial r_i}\right) = \left(1 + \left(\frac{dr}{dz}\right)^2\right)\frac{\partial r_i}{\partial t}(\rho \Delta H)$$

CLOSURE NEEDED FOR THE FREEZING RATE

Physically this depends on the temperature gradients at the interface

- Without a 2D simulation, one approach is to assume a fully developed profile in the solid layer

$$\frac{\partial}{\partial r}(rq) = 0 \quad \Rightarrow \quad -k_s \frac{dT_s}{dr} = \frac{C_1}{r}$$

- With the heat flux boundary condition, we have

$$C_1 = q''_{ws} R \quad \Rightarrow \quad \frac{dT_s}{dr} = \frac{-q''_{ws} R}{k_s r}$$

- If for the liquid, we assume the expression for heat transfer used earlier, the full equation to determine the rate of freezing is

$$\left(\frac{-q''_{ws} R}{r} - h_{int}(T_m - T_l) \right) = \left(1 + \left(\frac{dr_i}{dz} \right)^2 \right) \frac{\partial r_i}{\partial t} (\rho \Delta H)$$

ADDITIONAL RELATIONS

Areas and mass transfer rate are related to interface radius change

- Additional relations – many of the introduced variables are related simply through these relations. These first are simply geometric

$$A_l = \pi\left(r_i^2\right) \qquad A_s = \pi(R^2 - r_i^2)$$

$$\frac{\partial A_l}{\partial t} = \frac{\partial A_l}{\partial r_i}\frac{\partial r_i}{\partial t} = \pi(2r_i)\frac{\partial r_i}{\partial t} \qquad \frac{\partial A_s}{\partial t} = \frac{\partial A_s}{\partial r_i}\frac{\partial r_i}{\partial t} = \pi(-2r_i)\frac{\partial r_i}{\partial t} = -\frac{\partial A_l}{\partial t}$$

$$P_i = 2\pi r_i$$

- If assuming constant solid density, then the mass transfer rate is related to the radius by

$$\rho\frac{\partial r_i}{\partial t}(2\pi r_i) = -\sigma_A$$

SCOPE OF PROJECT FOR THE FUTURE

- Perform calculations with the analytical model and make comparisons to CFD
  - Determine suitability of model and path forward
    - If freezing closure model can be used with adjustments or
    - If a different, more empirical approach is required

- Regardless, currently a lot of work to do actually implementing capability in SAM

- Other students are working on experiments – validation is key
  - Static experiments with FLiBe currently under way in glove boxes
  - Pipe experiments being designed