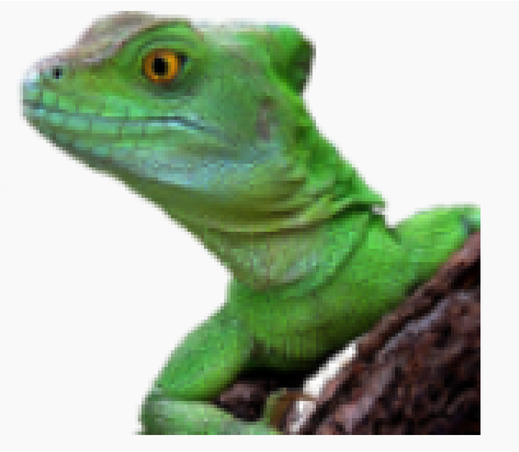
# Numerical modelling of binary solidification by using Basilisk

Basilisk (Gerris) Users' Meeting 2023  PARIS

**Jie ZHANG**, Zhong-Han XUE, Bo-Lin WEI, Ming-Jiu Ni



School of Aerospace, Xi'an Jiaotong University

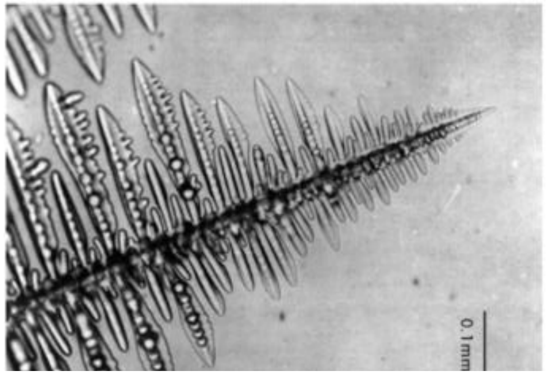
7th, July 2023

# Outline

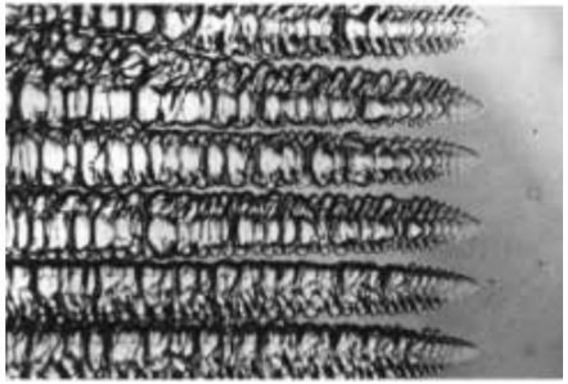
- **Background**
- **Numerical Methods**
- **Validation tests**

# Research Background

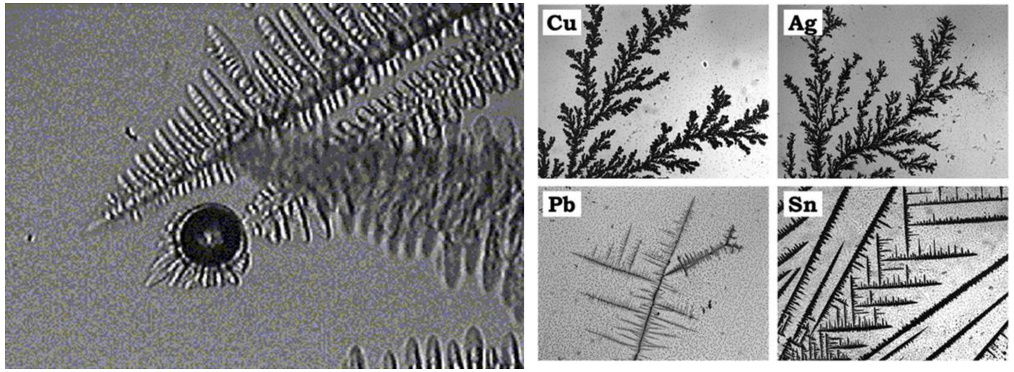
- **Metallurgy**： binary solidification and dendritic growth（e.g. Ni-Cu alloy）




*Jackson et al. 2006*



*Travedi 1984*




- **Experimental study**
  - ☐ Liquid metal is **opaque**
  - ☐ Difficult to reveal the **mechanisms** controlling the solidification

- **Present study**
  - ☐ Developing **accurate** and **sharp** numerical methods
  - ☐ Modelling the binary solidification

# Research Background —— Physical model

- Temperature and concentration diffusion equations

$$\frac{\partial T_{l,s}}{\partial t} = \alpha_{l,s}\nabla^2 T_{l,s} \quad \forall \mathbf{x} \in \Omega_{l,s}$$

$$\frac{\partial Y_{l,s}}{\partial t} = D_{l,s}\nabla^2 Y_{l,s} \quad \forall \mathbf{x} \in \Omega_{l,s}$$

- Gibbs-Thomson equation

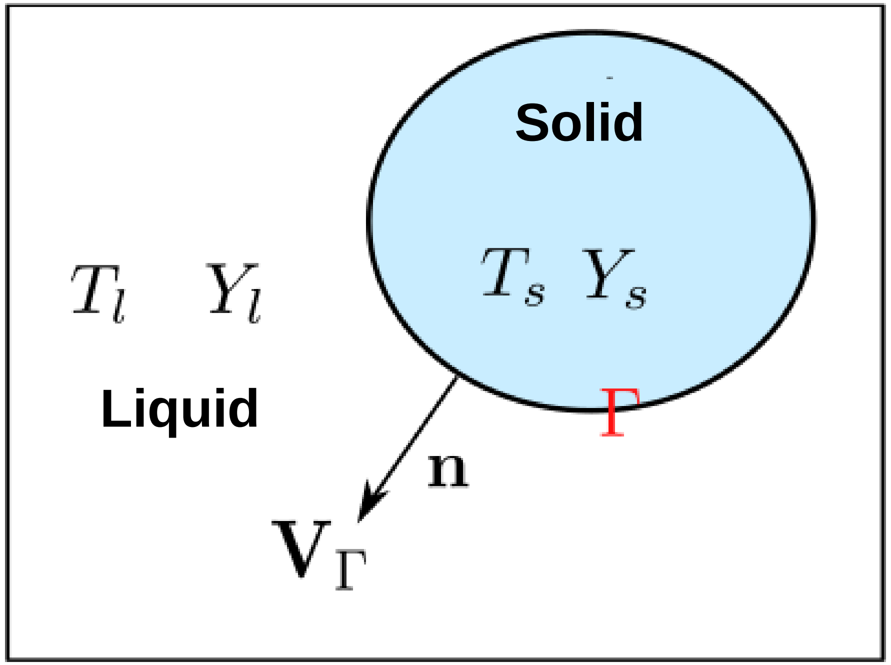
$$T_\Gamma = T_m + m_L Y_l|_\Gamma - \epsilon_\kappa \kappa - \epsilon_V V_\Gamma \quad \forall \mathbf{x} \in \Gamma$$

- Stefan equation

$$\rho_s L V_\Gamma = (k_s \nabla T_s - k_l \nabla T_l) \cdot \mathbf{n} \quad \forall \mathbf{x} \in \Gamma$$

- Solute rejection relation

$$(1 - k_p) Y_l V_\Gamma = (D_s \nabla Y_s - D_l \nabla Y_l) \cdot \mathbf{n} \quad \forall \mathbf{x} \in \Gamma$$

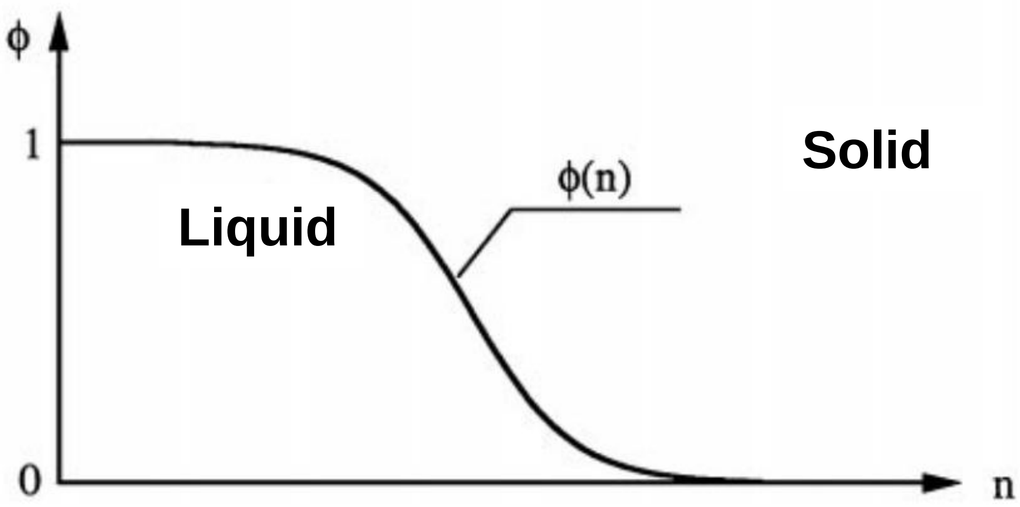



Problem：

How to design sharp schemes to capture the jump conditions across the interface?

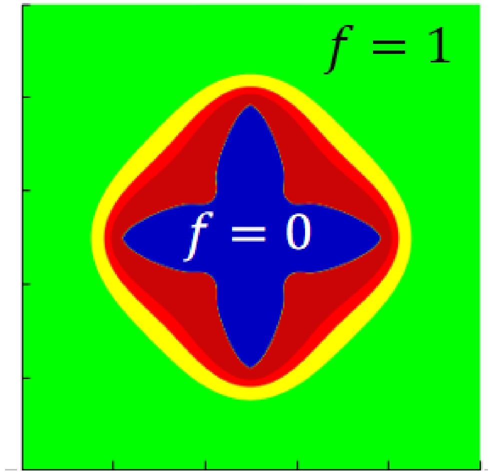
# Research Background —— Diffused interface method

## Phase-Field Method




$$\tau \partial_t \phi = W^2 \nabla^2 \phi - \partial_\phi \left[ f(\phi) + \lambda g(\phi) u \right]$$
$$\partial_t c + \vec{\nabla} \cdot \vec{j} = 0,$$

## Enthalpy-Based Method

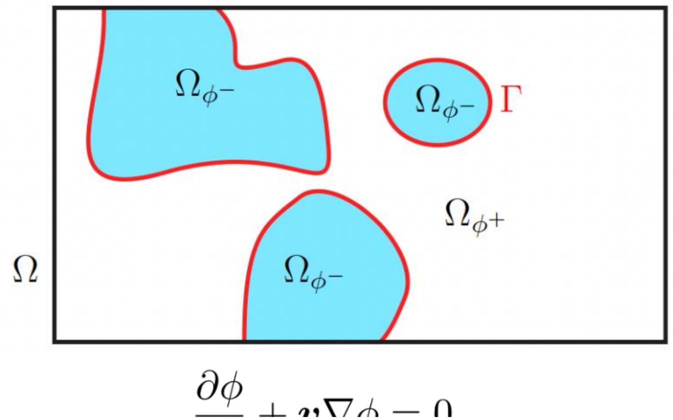



$$\frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T = \kappa \nabla^2 T - \frac{L}{c_p} \frac{\partial f}{\partial t}$$

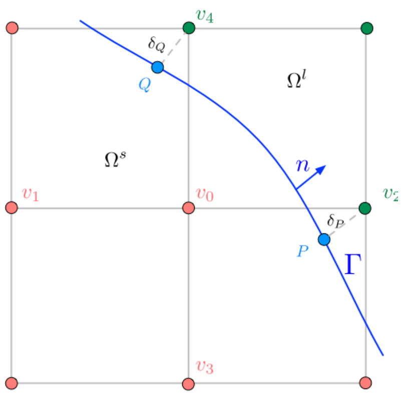
- Interface is represented by the smooth function $\phi$ or $T$, with source terms
- No need to impose jump conditions at the solid-liquid interface

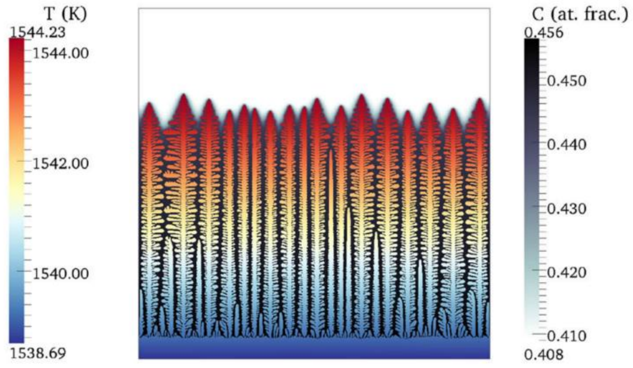
The underlying physics near the interface is **undermined**!

# Research Background —— Sharp interface method

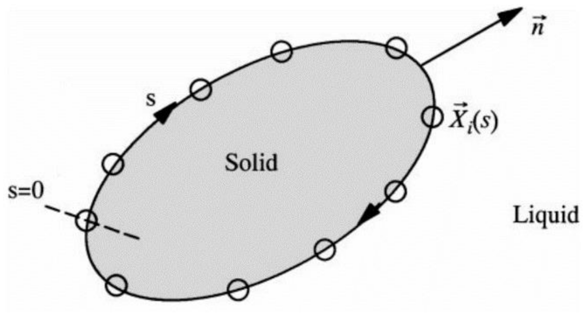


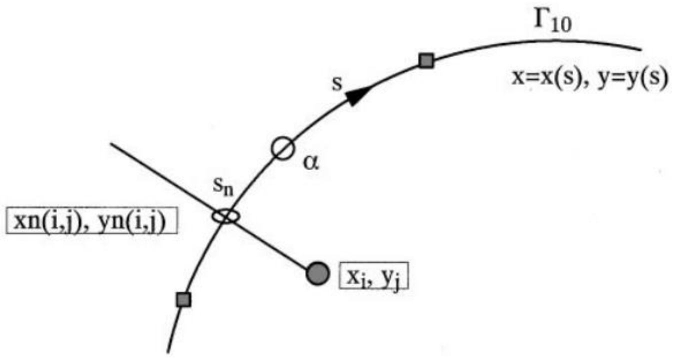

*Theillard et al. 2015 JSC*

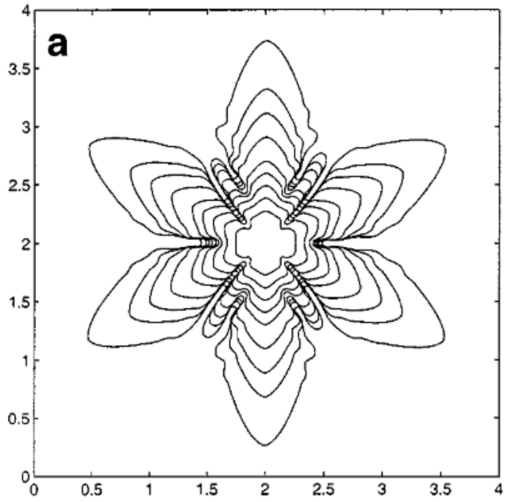



Level-set Method + Ghost Fluid Method



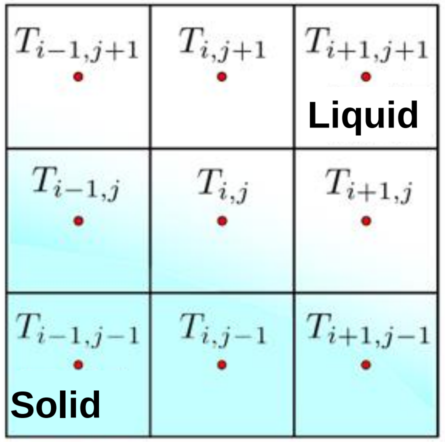




*Udaykumar et al. 1999 JCP*

Front-Tracking Method + Immersed Solid Method

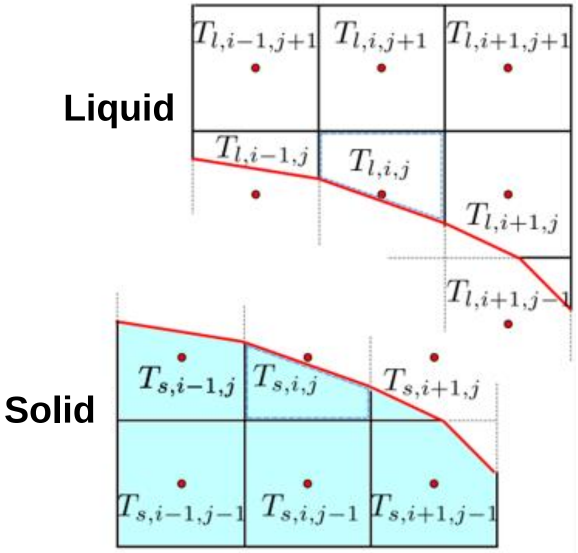
# Research Background —— Sharpness or Smoothness




## Diffused interface method (one-fluid model)

$$(\partial_t + \boldsymbol{u} \cdot \nabla) T = \alpha \nabla^2 T + St^{-1} \ddot{m} \quad \boldsymbol{x} \in \Omega.$$

- Solve one governing equation in the whole domain
- The jump conditions are transferred into source terms




## Sharp interface method (two-fluid model)

$$(\partial_t + \boldsymbol{u} \cdot \nabla) T^\ell = \alpha^\ell \nabla^2 T^\ell \quad \boldsymbol{x} \in \Omega^\ell$$

$$\partial_t T^s = \alpha^s \nabla^2 T^s \quad \boldsymbol{x} \in \Omega^s$$

$$\rho_s L V_\Gamma = ((k\nabla T)^s - (k\nabla T)^\ell) \cdot \boldsymbol{n} \quad \boldsymbol{x} \in \Gamma$$

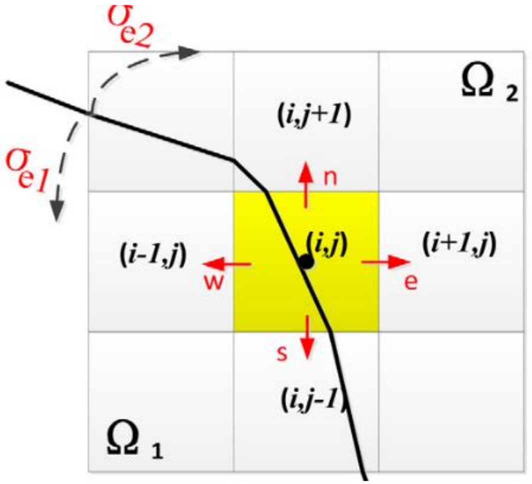
- The jump conditions are imposed sharply at the interface

# Objectives

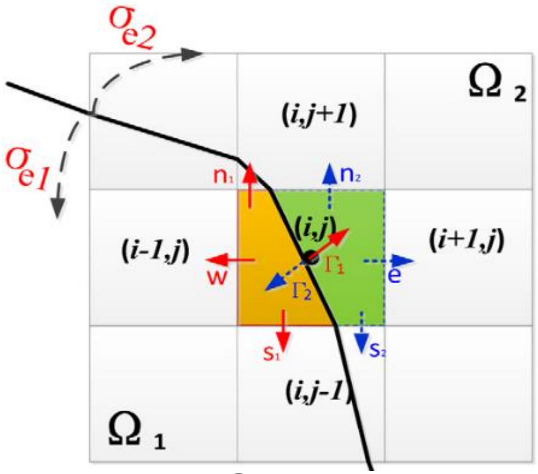
To develop a VOF-based sharp interface method for modelling binary solidification

Advantages of the VOF-based methods:

- Conservativeness
- Sharp interface representation
- Combined with EBM to impose sharp jump conditions




**Original VOF method**




**VOF + EBM**

# Outline

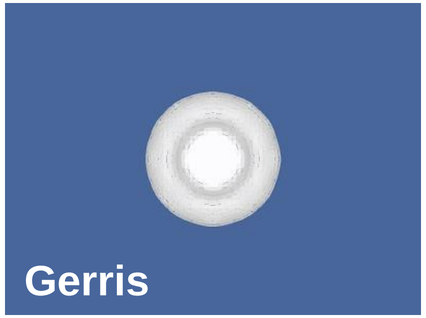
- **Background**
- **Numerical Methods**
- **Validation tests**

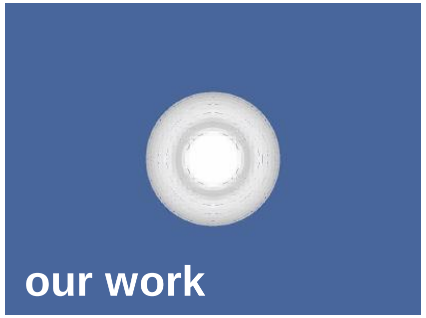
# Outline

**Our recent progress of code implementation Basilisk/Gerris**

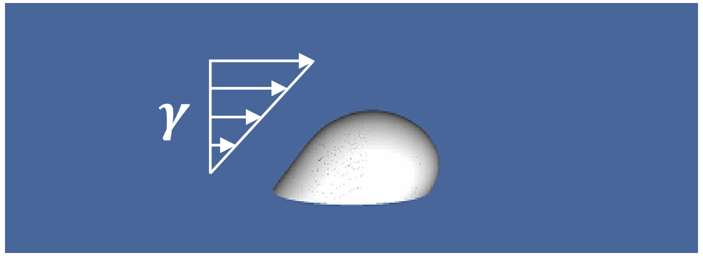
# Numerical methods

## 1. 3D Contact angle



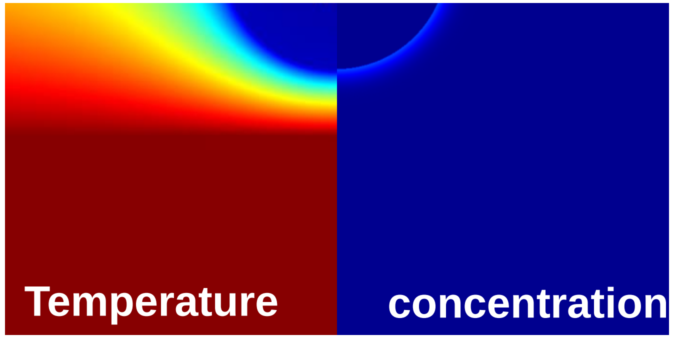



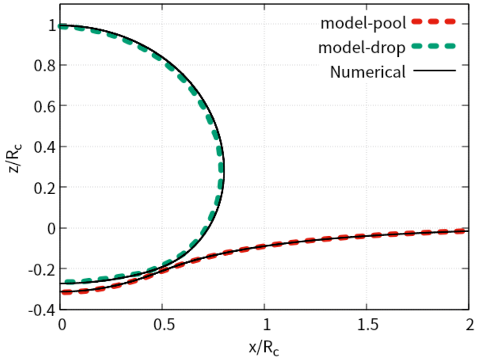

**top view of the droplet impacting**



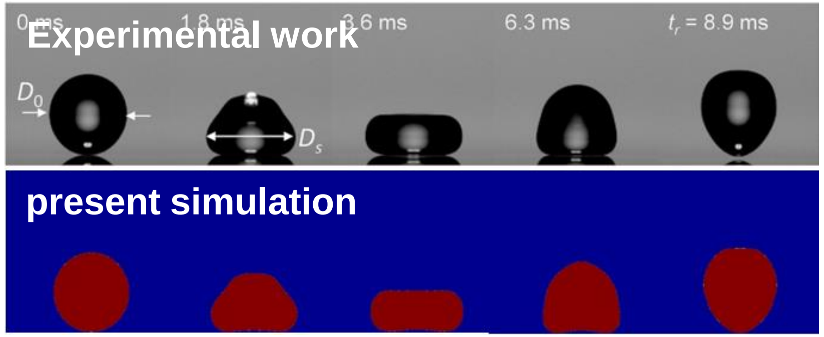

**Contact line hysteresis**

## 2. Evaporation flows



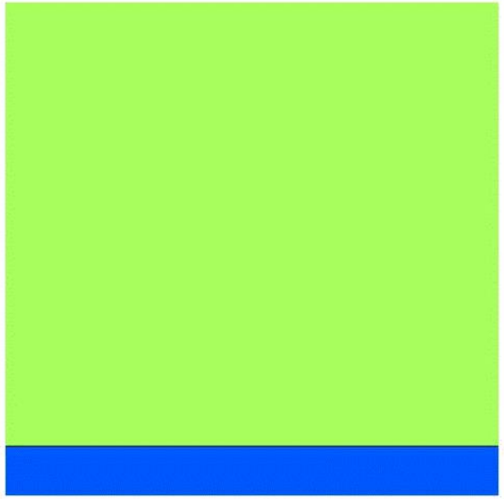

**Impacting onto hot pool: DNS of Leidenfrost effect**




**Leidenfrost effect onto solid substrate**

# Outline

## 3. Binary solidification



# Physical model

- Diffusion equations

$$\frac{\partial T_{l,s}}{\partial t} = \alpha_{l,s}\nabla^2 T_{l,s} \quad \mathbf{x}\in\Omega_{l,s}, \qquad (1)$$

$$\frac{\partial Y_{l,s}}{\partial t} = D_{l,s}\nabla^2 Y_{l,s} \quad \mathbf{x}\in\Omega_{l,s}. \qquad (2)$$

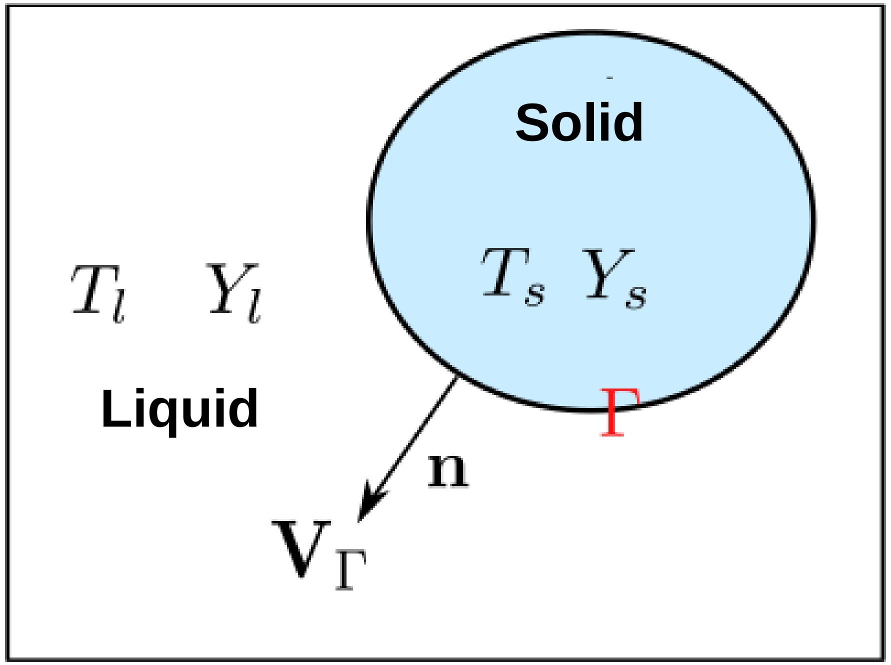
- Jump conditions

$$L_H V_\Gamma = (k_s\nabla T_s - k_l\nabla T_l)\cdot\mathbf{n}, \qquad (3)$$

$$(1-k_p)Y_{l,\Gamma}V_\Gamma = -(D_l\nabla Y_l)\cdot\mathbf{n}. \qquad (4)$$

- Gibbs-Thomson equation

$$T_\Gamma = \underbrace{T_m}_{\text{Melting Temp.}} + \underbrace{m_L Y_\Gamma}_{\text{Solute Conc.}} + \underbrace{\epsilon_c\kappa}_{\text{Capillary Undercooling}} + \underbrace{\epsilon_v V_\Gamma}_{\text{Kinetic Undercooling}}. \qquad (5)$$

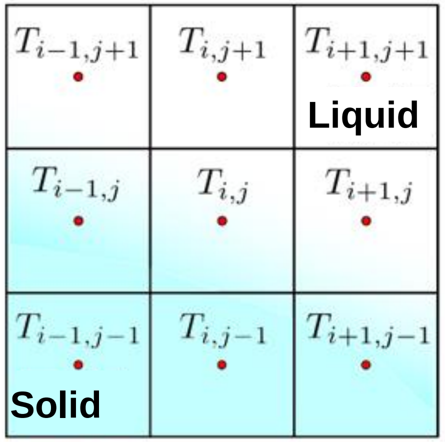



**Difficulty:**

Sharp jump conditions

Coupling of T and Y at interface

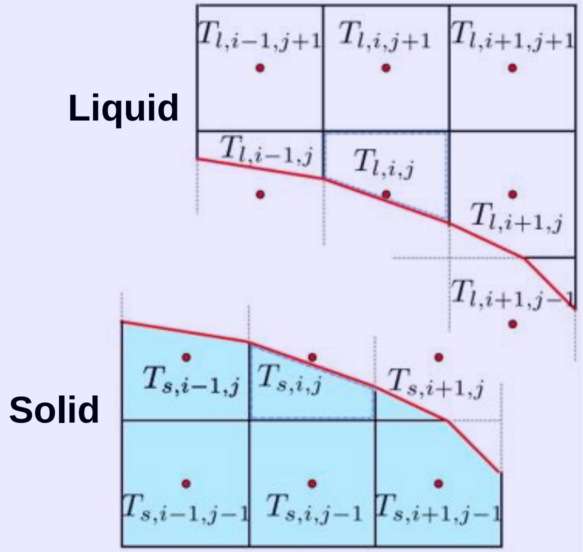
# Sharp interface method




Diffused interface method (one-fluid model)

$$(\partial_t + \boldsymbol{u} \cdot \nabla) T = \alpha \nabla^2 T + St^{-1} \ddot{m} \quad \boldsymbol{x} \in \Omega.$$

- Solve one governing equation in the whole domain
- The jump conditions are transferred into source terms




**Sharp interface method (two-fluid model)**

$$(\partial_t + \boldsymbol{u} \cdot \nabla) T^\ell = \alpha^\ell \nabla^2 T^\ell \quad \boldsymbol{x} \in \Omega^\ell$$

$$\partial_t T^s = \alpha^s \nabla^2 T^s \quad \boldsymbol{x} \in \Omega^s$$

$$\rho_s L V_\Gamma = ((k \nabla T)^s - (k \nabla T)^\ell) \cdot \boldsymbol{n} \quad \boldsymbol{x} \in \Gamma$$

- The jump conditions are imposed sharply at the interface

# Global procedures

For binary fluid, the diffusivity of solute is much smaller than that of temperature.
To avoid numerical oscillation, the front velocity need to be computed according to the solute.

**Velocity >> Concentration >> Temperature >> Velocity**

1. $Y^{n+1} \leftarrow V_\Gamma^n$ (Robin BC)

$$(1-k_p)Y_l V_\Gamma = -D_l \nabla Y_l \cdot \mathbf{n}.$$

2. $T^{n+1} \leftarrow Y^{n+1}$ (Dirichlet BC)

$$T_\Gamma = T_m + m_L Y_\Gamma + \epsilon_c(\theta)\kappa + \epsilon_v V_\Gamma.$$

3. $V^{n+1} \leftarrow T^{n+1}$ (Interpolation)

$$V_\Gamma = \frac{1}{L_H}(k_s \nabla T_s - k_l \nabla T_l) \cdot \mathbf{n}.$$

**Velocity >> Temperature >> Concentration >> Velocity**

1. $T^{n+1} \leftarrow V_\Gamma^n$ (Jump condition)

$$L_H V_\Gamma = (k_s \nabla T_s - k_l \nabla T_l) \cdot \mathbf{n}.$$

2. $Y^{n+1} \leftarrow T^{n+1}$ (Dirichlet BC)

$$T_\Gamma = T_m + m_L Y_\Gamma + \epsilon_c(\theta)\kappa + \epsilon_v V_\Gamma.$$

3. $V^{n+1} \leftarrow Y^{n+1}$ (Interpolation)

$$V_\Gamma = -\frac{D_l}{(1-k_p)Y_l} \nabla Y_l \cdot \mathbf{n}.$$

# Numerical method – Flux jump conditions

- Given jump conditions from Stefan condition

$$[\vartheta\nabla\chi]_\Gamma = \Big((\vartheta\nabla_\Gamma\chi)_s - (\vartheta\nabla_\Gamma\chi)_l\Big)\cdot \boldsymbol{n} = \gamma \quad \forall \boldsymbol{x}\in\Gamma$$

$$[\chi]_\Gamma = 0 \quad \forall \boldsymbol{x}\in\Gamma$$

- Finite volume method + Divergence theorem

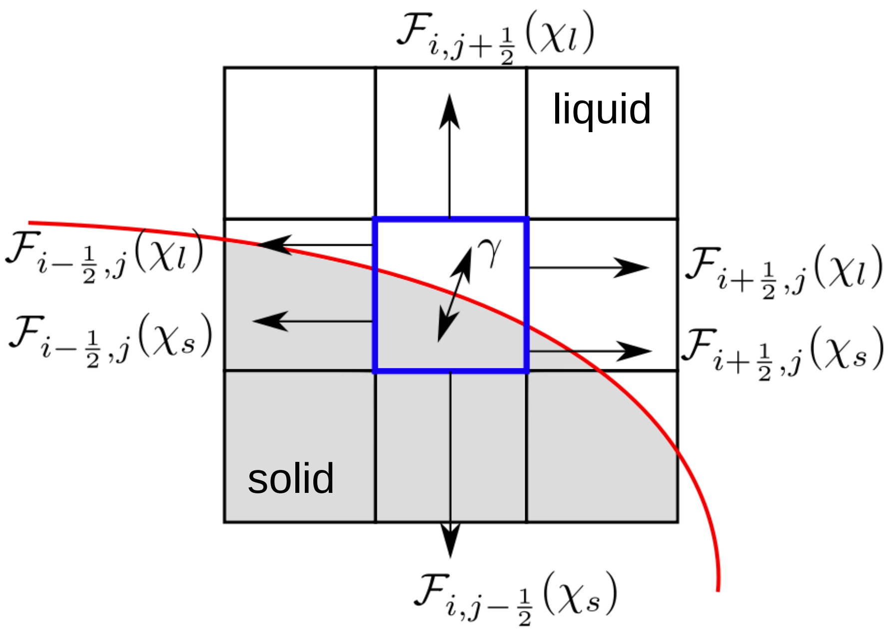
$$\int_{\Omega_{i,j}\cap\Omega_l} \nabla_c\cdot\Big((\vartheta_l)_f\nabla_f\chi_l\Big)^{n+1}\mathrm{d}V + \int_{\Omega_{i,j}\cap\Omega_s} \nabla_c\cdot\Big((\vartheta_s)_f\nabla_f\chi_s\Big)^{n+1}\mathrm{d}V$$

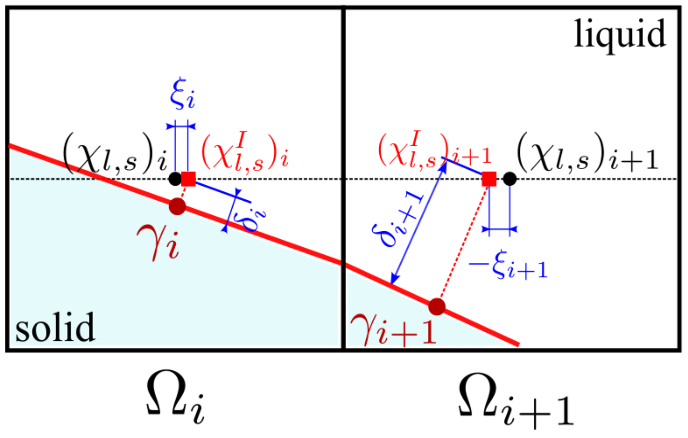
$$= \underbrace{\left(\vartheta_l S_l\frac{\partial\chi_l}{\partial \boldsymbol{n}}\right)_{i+\frac12,j} + \left(\vartheta_l S_l\frac{\partial\chi_l}{\partial \boldsymbol{n}}\right)_{i-\frac12,j} + \left(\vartheta_s S_s\frac{\partial\chi_s}{\partial \boldsymbol{n}}\right)_{i+\frac12,j} + \left(\vartheta_s S_s\frac{\partial\chi_s}{\partial \boldsymbol{n}}\right)_{i-\frac12,j}}_{\text{partial face}}$$

$$+ \underbrace{\left(\vartheta_l S_l\frac{\partial\chi_l}{\partial \boldsymbol{n}}\right)_{i,j+\frac12} + \left(\vartheta_s S_s\frac{\partial\chi_s}{\partial \boldsymbol{n}}\right)_{i,j-\frac12}}_{\text{full face}} + \underbrace{\gamma S_\Gamma}_{\text{reconstructed interface}}$$

- Relation between cell centered values for two phases

$$(\chi_l)_i - (\chi_s)_i = \frac{(\Delta+\xi_{i+1})\left(\frac{\delta\gamma}{\vartheta}\right)_i - \xi_i\left(\frac{\delta\gamma}{\vartheta}\right)_{i+1}}{\xi_i - \xi_{i+1} - \Delta} + O(\Delta^2)$$



# Outline

- **Background**
- **Numerical Methods**
- **Validation tests**

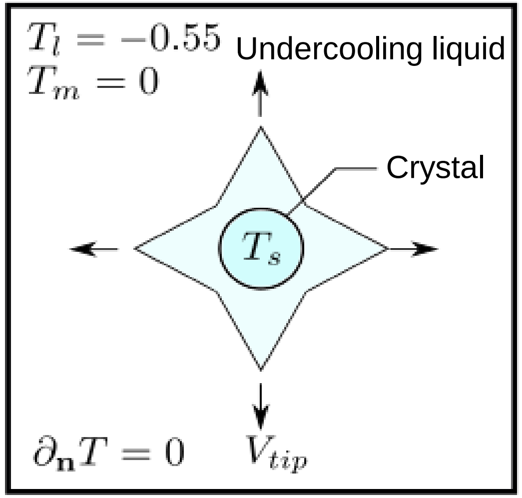
# Numerical tests – 2D unstable solidification

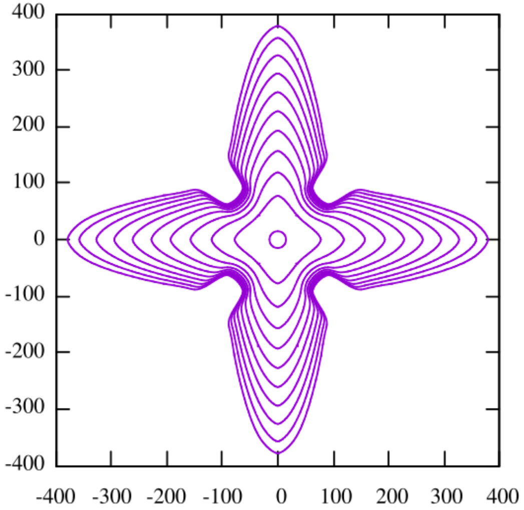
- Anisotropic capillary coefficient

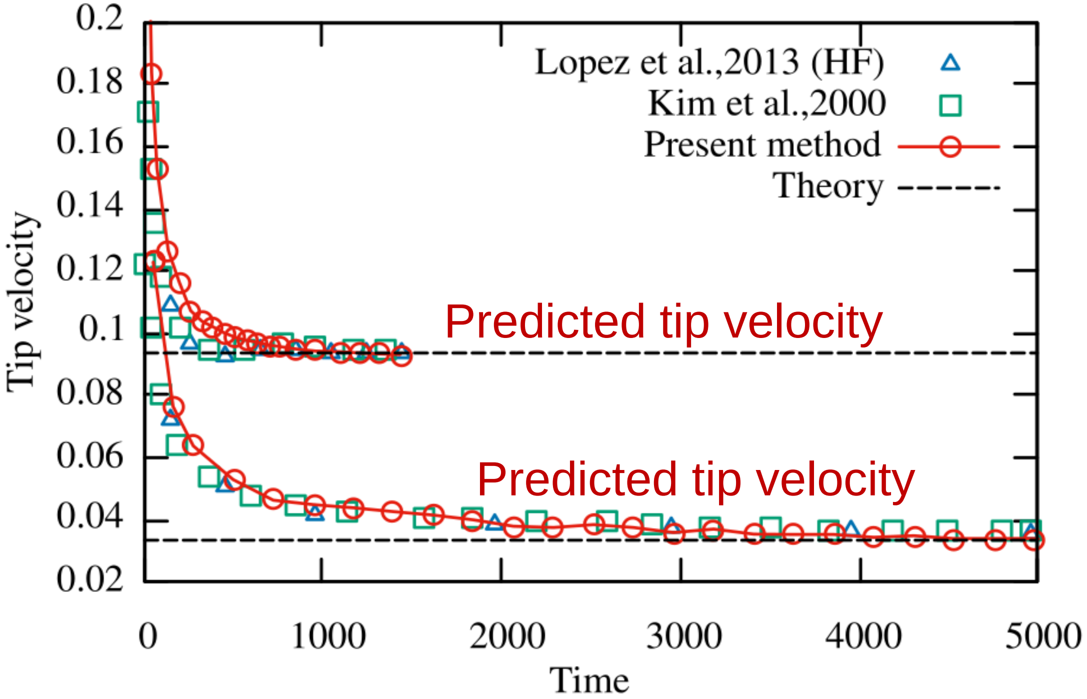
$$\epsilon_\kappa = 0.5(1 - 0.75\cos 4\theta)$$

- Predicted tip velocity

$$V_{tip} = \begin{cases} 0.034, & T_l = -0.55 \\ 0.0935 & T_l = -0.65 \end{cases}$$



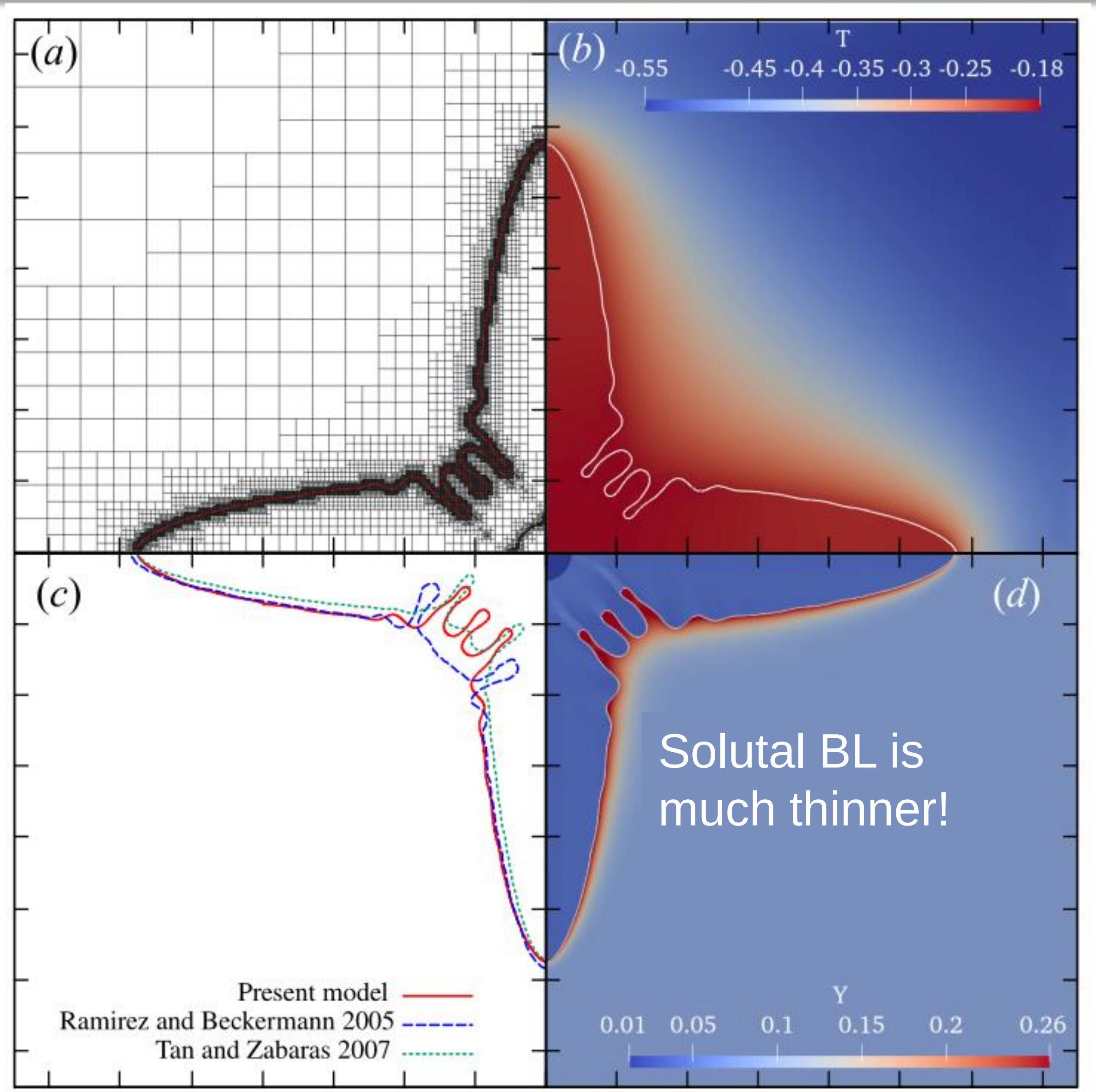



# Numerical tests – 2D unstable binary solidification

Adaptive mesh

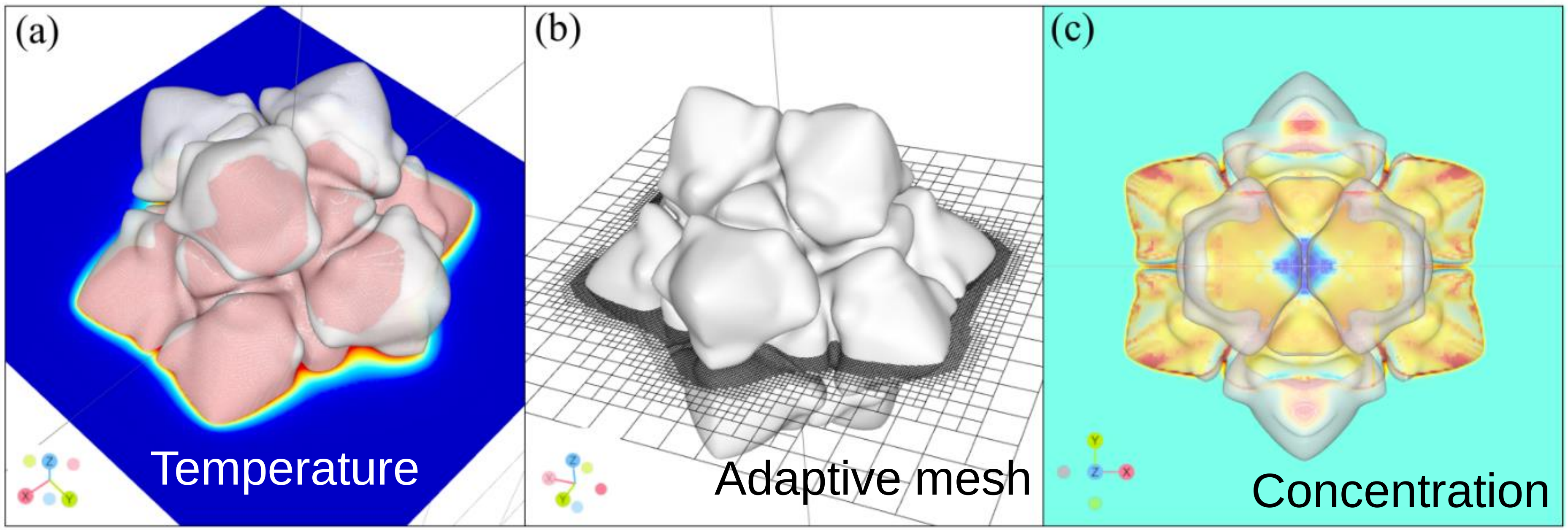
Temperature field

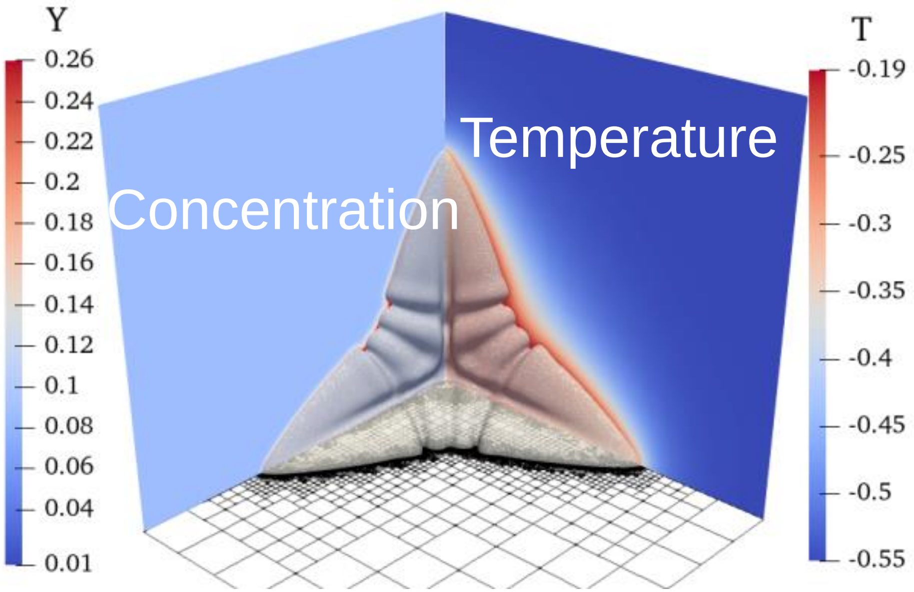


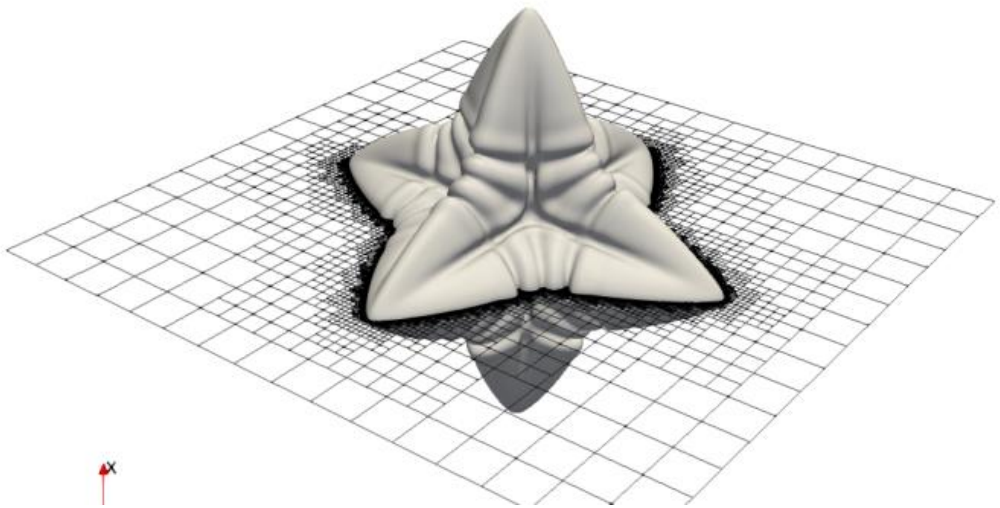

Comparison with other works

Concentration field

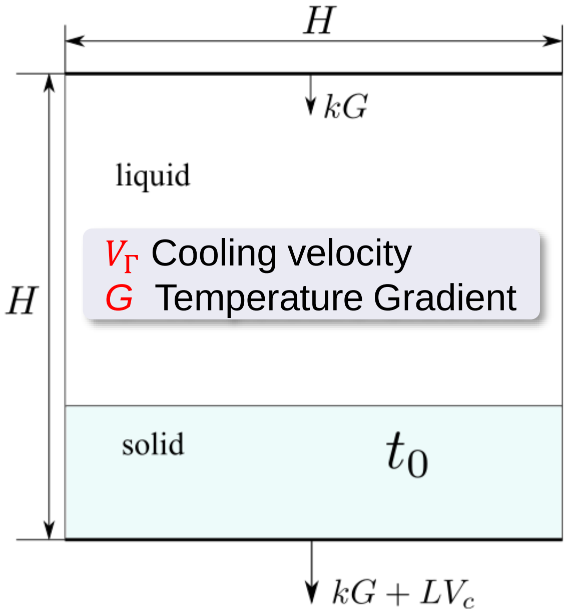
# Numerical tests – 3D unstable binary solidification



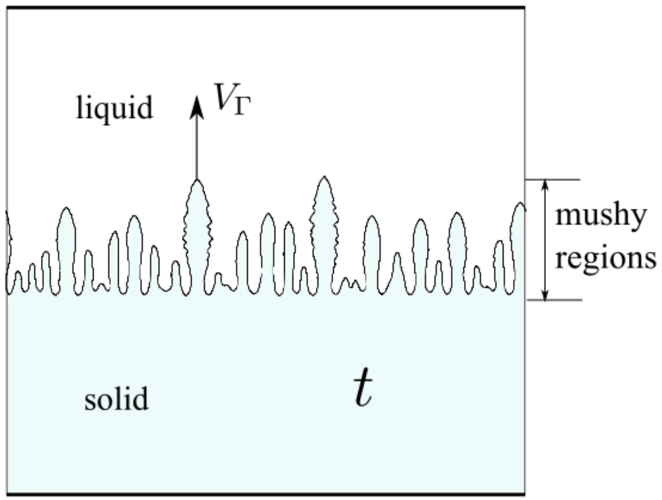



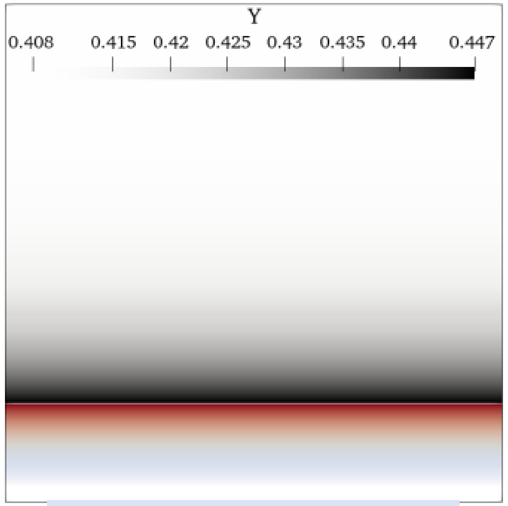
# Numerical tests – Binary solidification of a Ni-Cu alloy



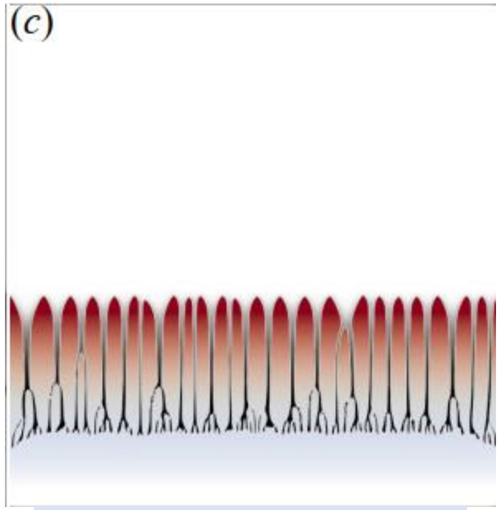

Binary solidification

$Le = 10^4$

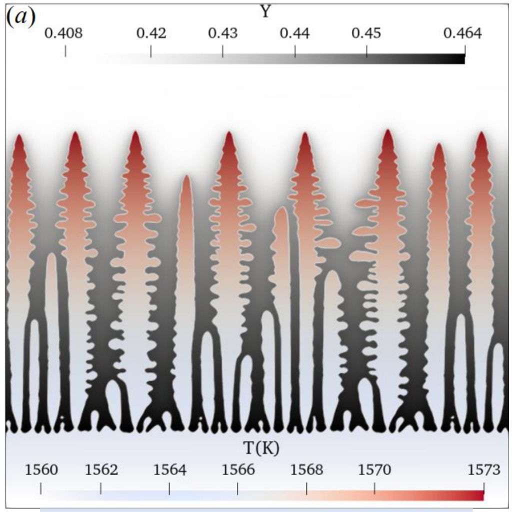



Planar growth



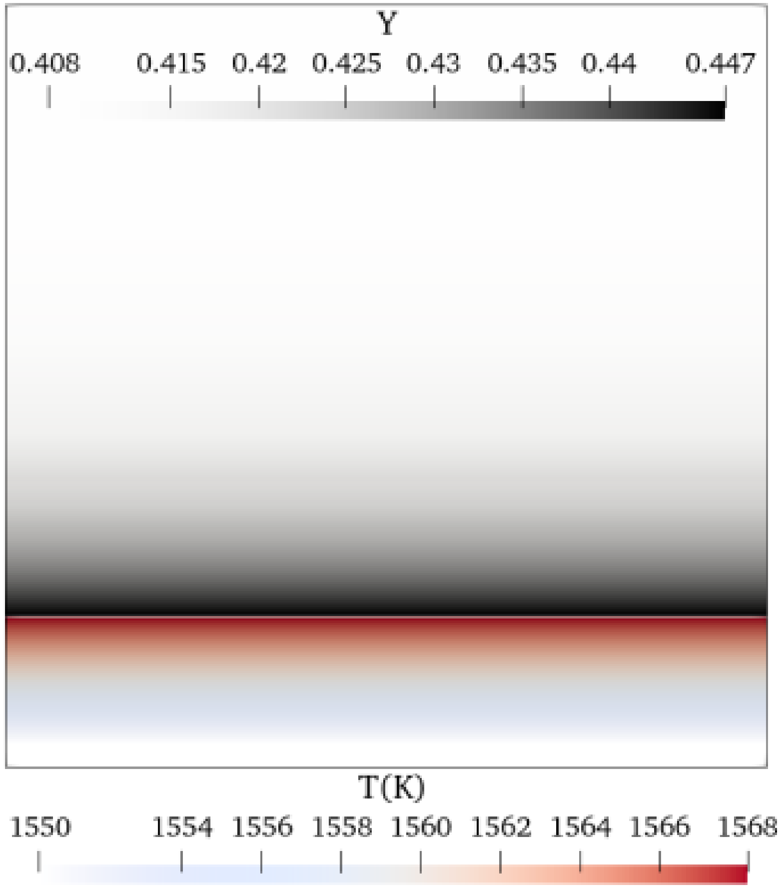

Cellular growth

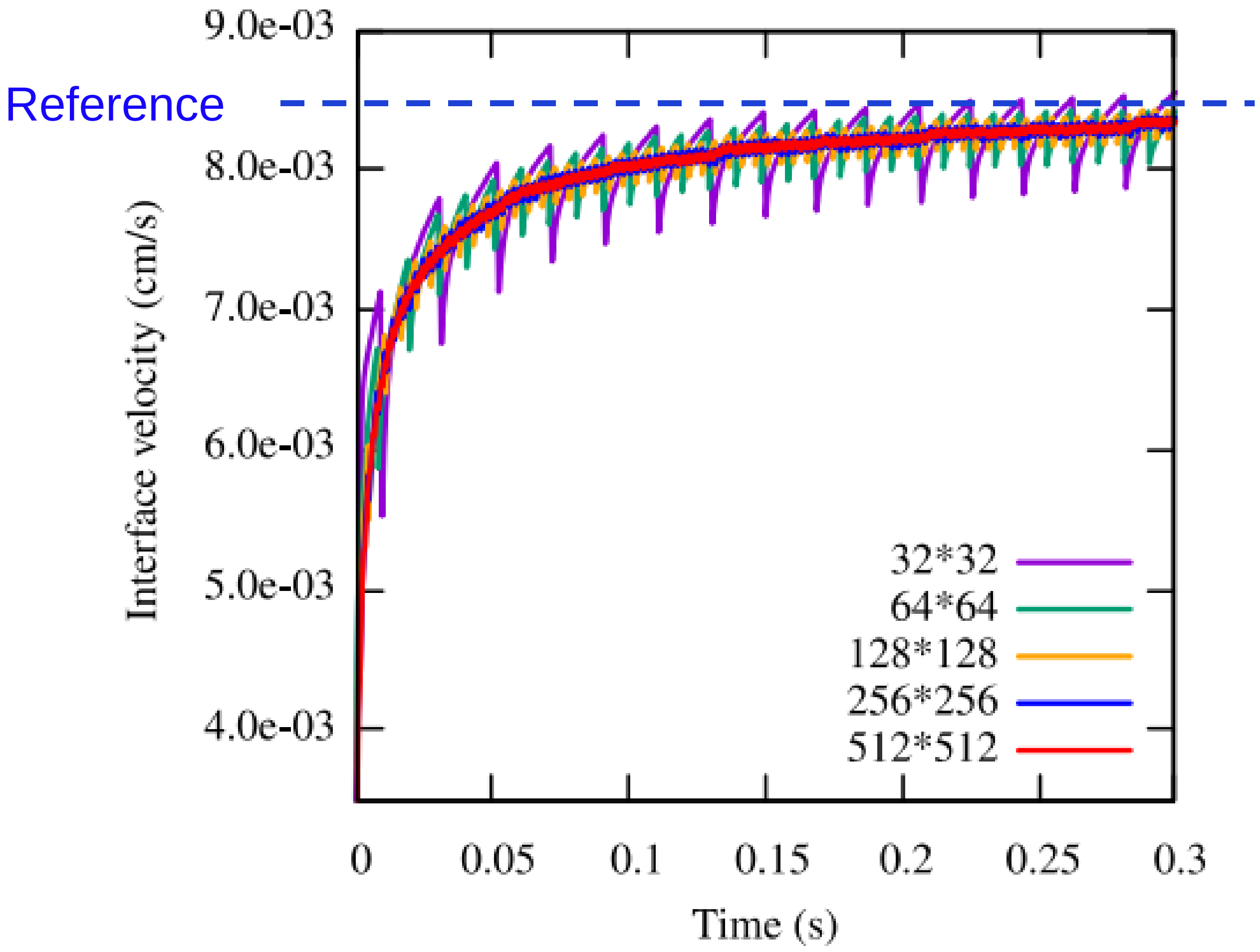



Dendritic growth

# Numerical tests – Binary solidification of a Ni-Cu alloy

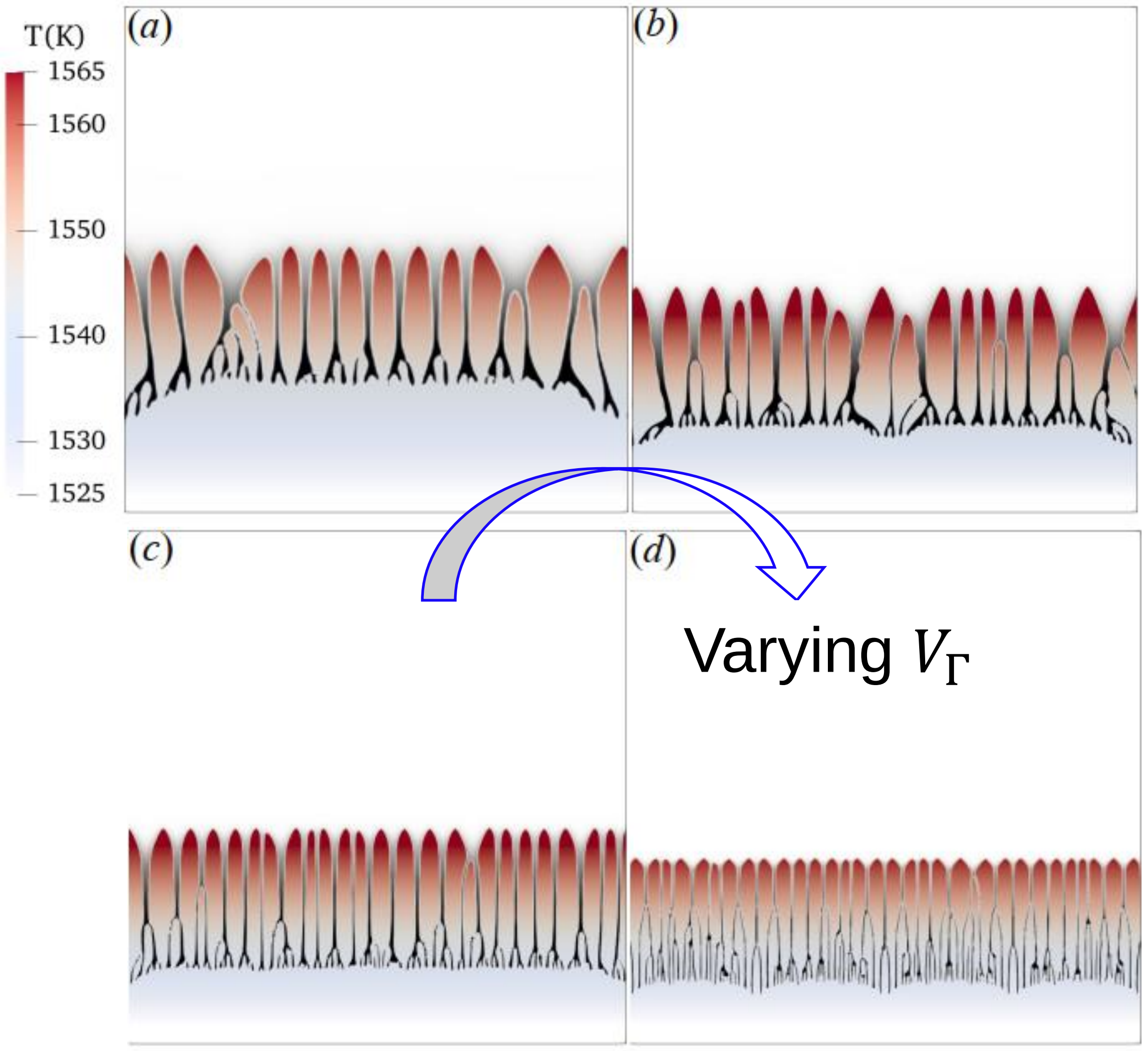
## Planar growth (stable)

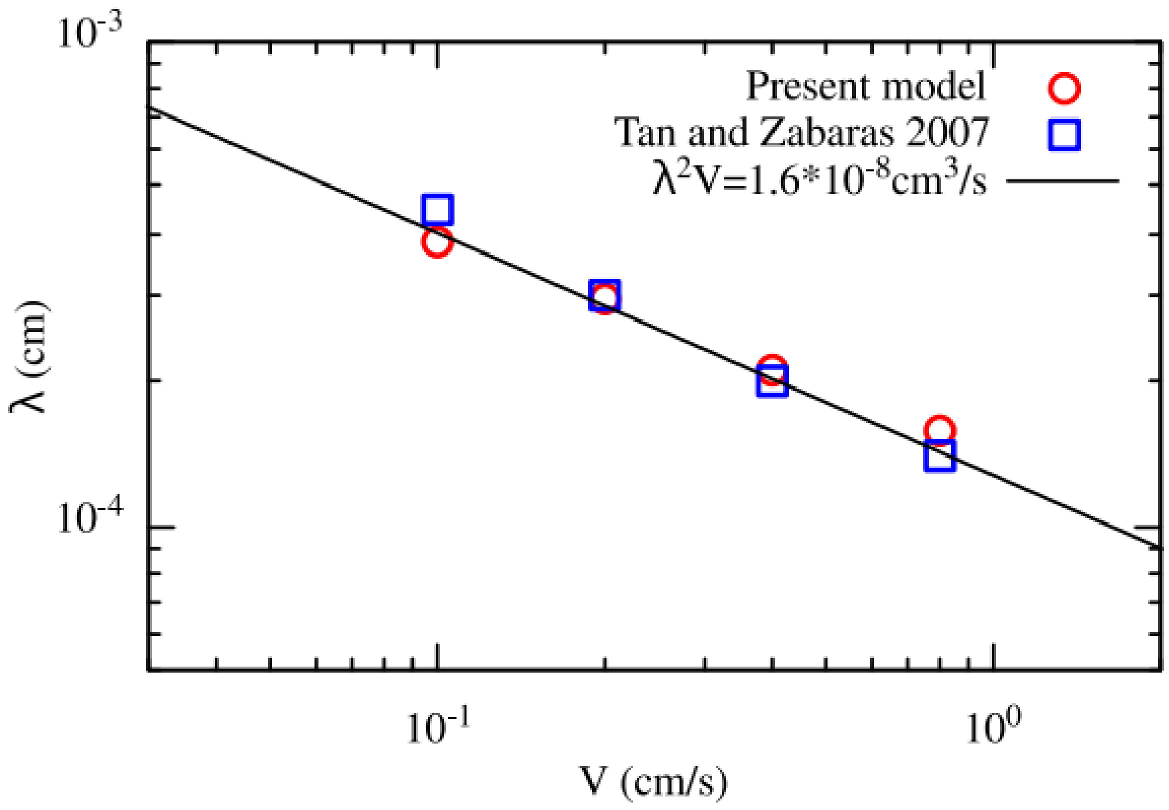


# Numerical tests – Binary solidification of a Ni-Cu alloy

## Cellular growth (unstable)

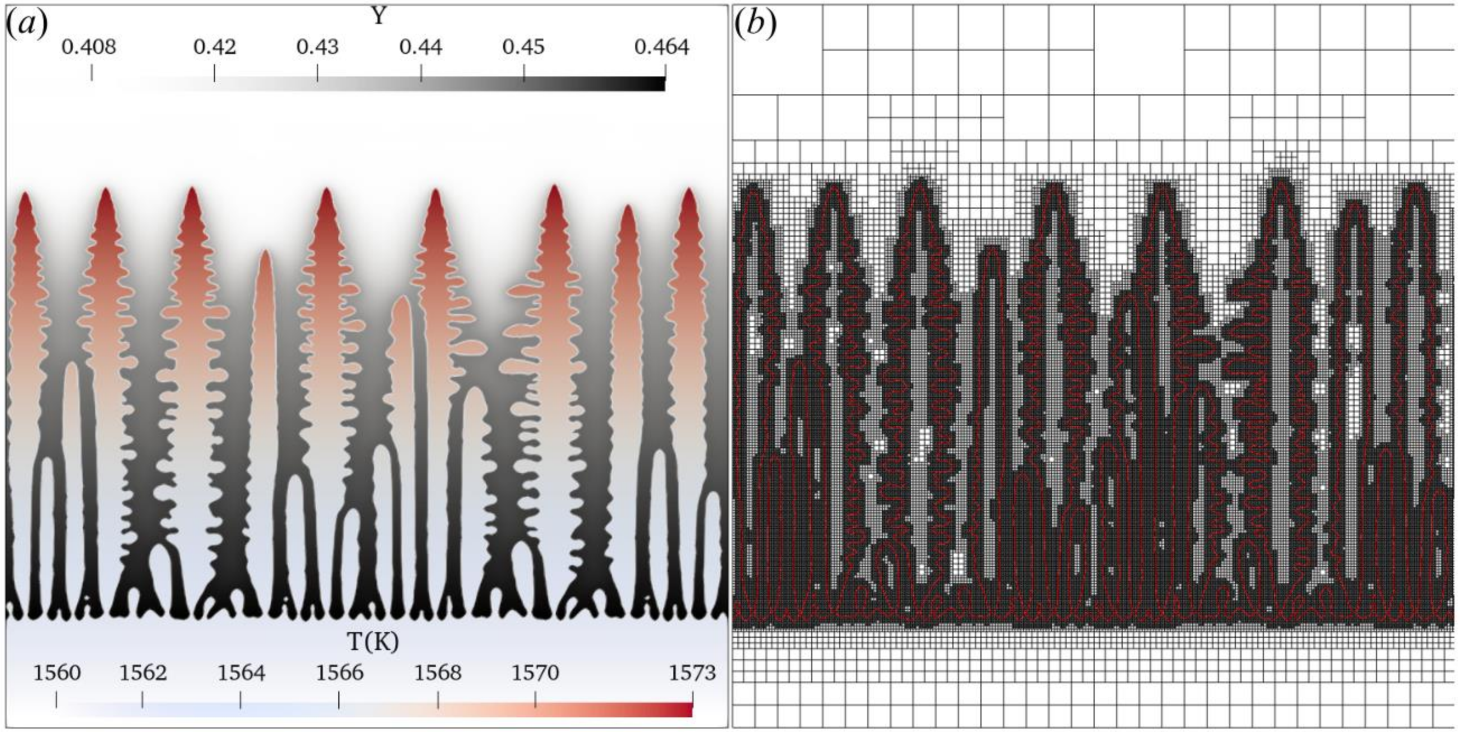


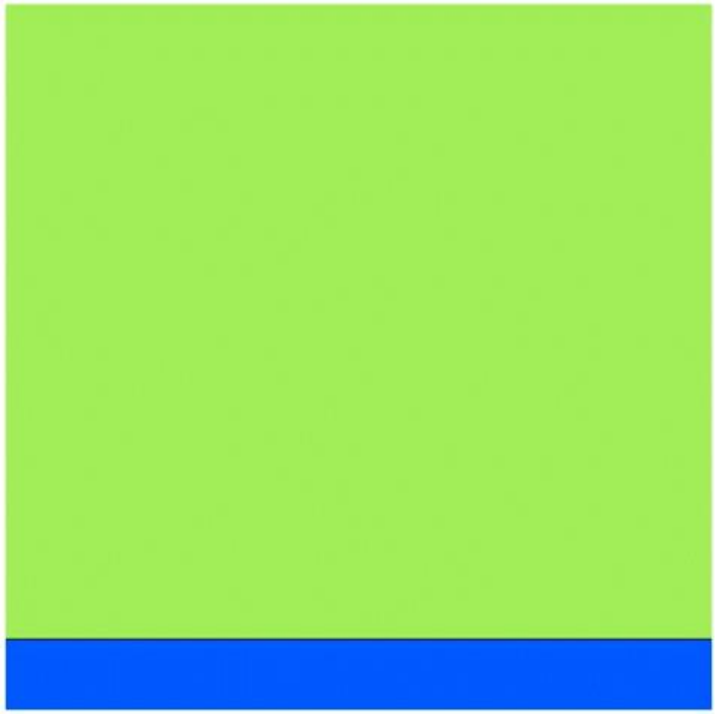




Arm spacing

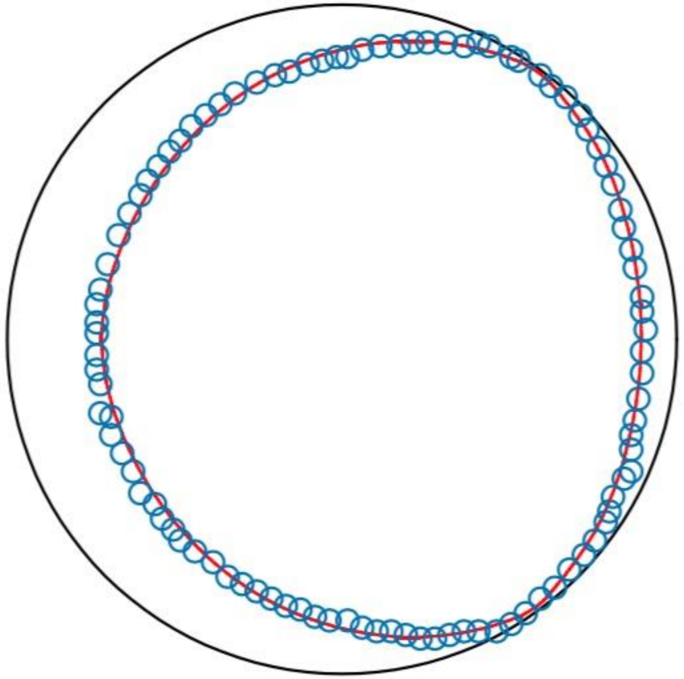
# Numerical tests – Binary solidification of a Ni-Cu alloy

## Dendritic growth (unstable)



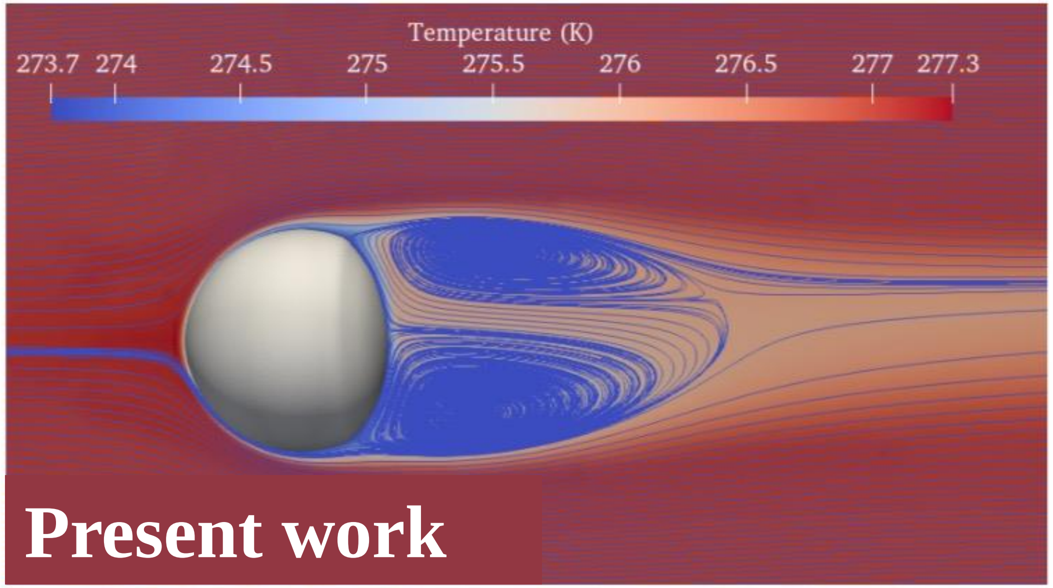
# Numerical tests – Melting ice sphere under forced convection

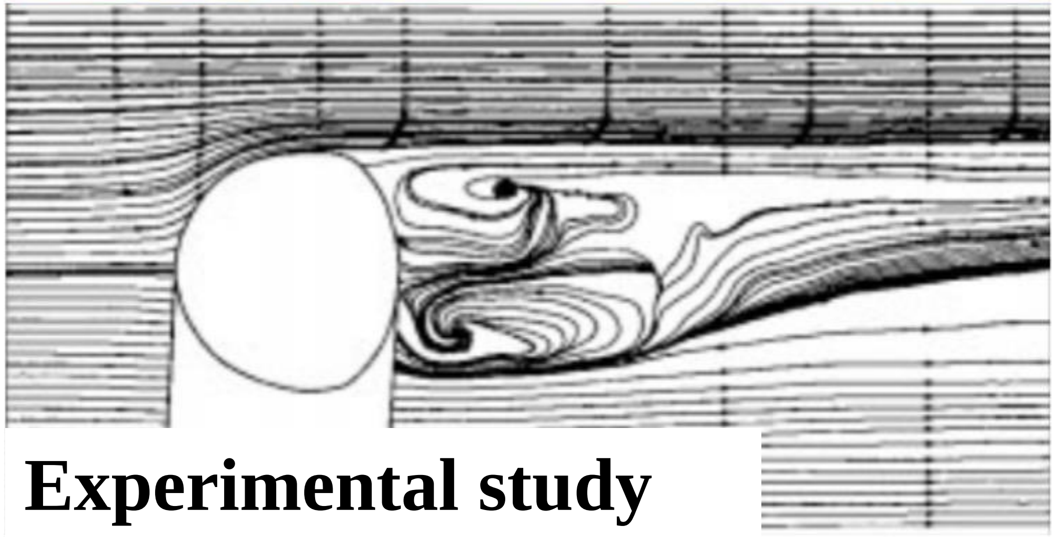


**Black solid line**: Initial interface

**Red Solid line**: Present work

**Unfilled points**: Experimental work

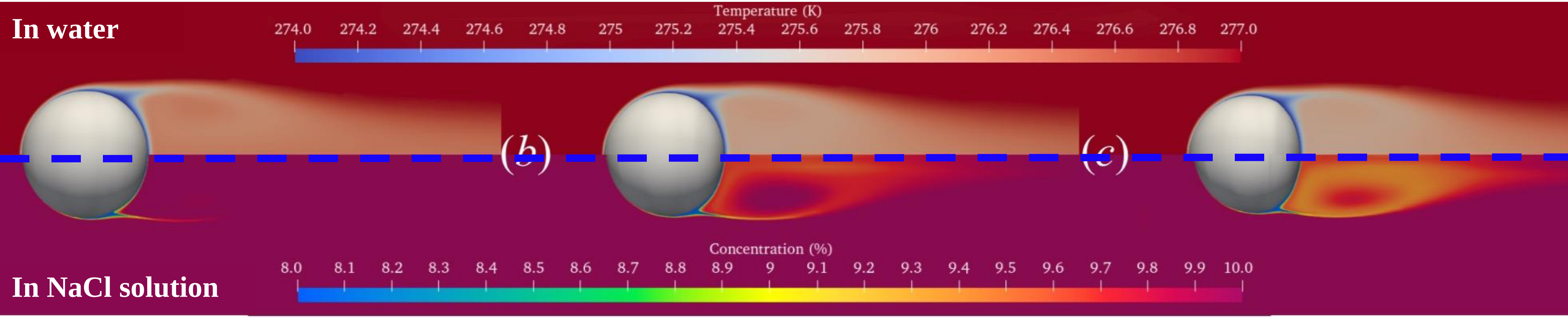


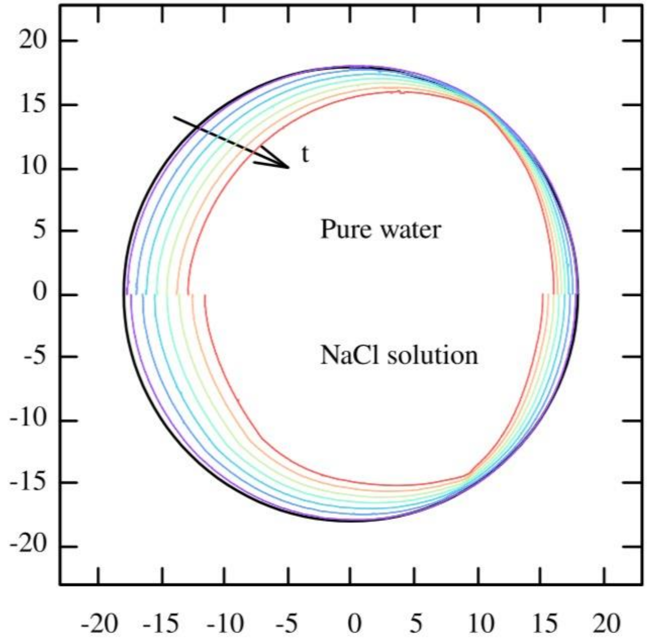

Y. L. Hao, Y.-X. Tao, Journal of heat transfer, 2001

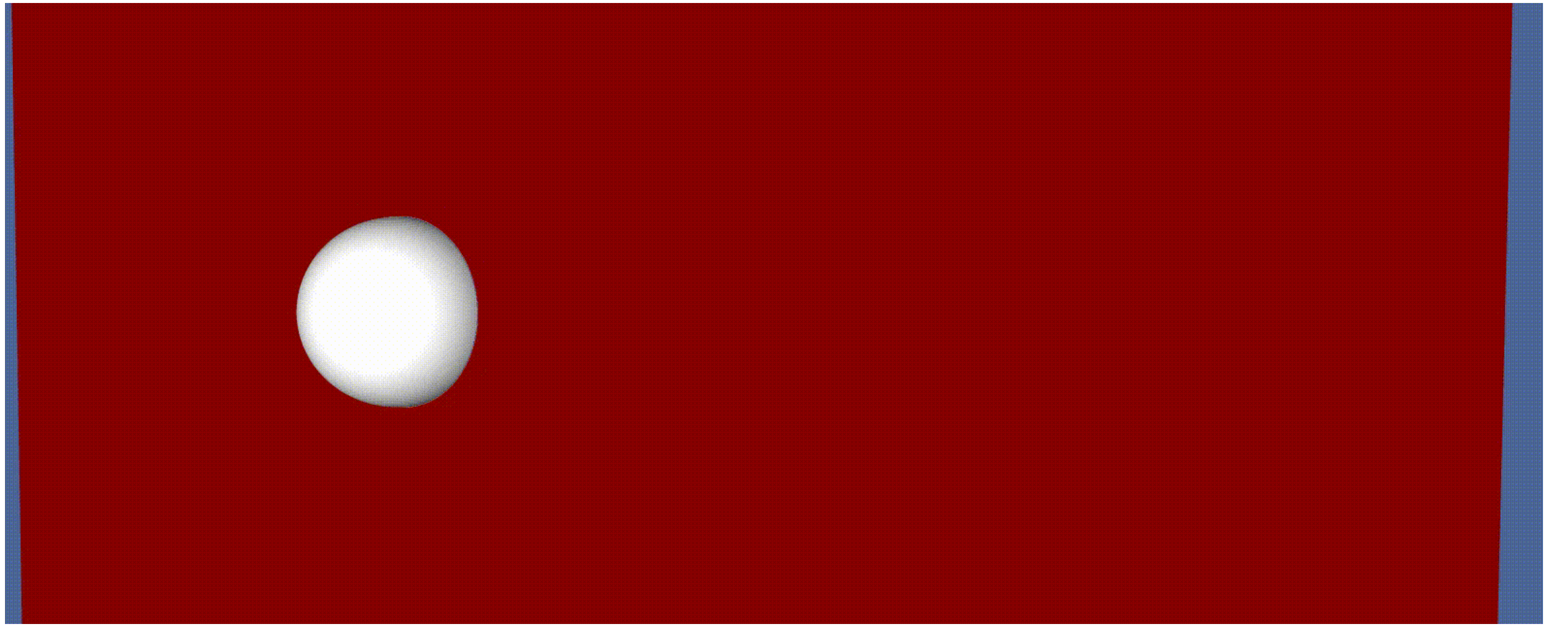
- Melting in 2.5°C water
- Melting in 2.5°C NaCl solution

# Numerical tests – Melting ice sphere under forced convection




$t = 144$ s  $t = 324$ s  $t = 576$ s



# Conclusion

- VOF-based sharp method for modelling binary solidification;
- Sharp flux jump of temperature at the solid-liquid interface;
- Unstable binary dendritic growth with high Lewis number;
- Powerful numerical tools in metallurgic applications

# Acknowledgement