New Perspectives in Modelling Heat Transfer and Multiphase Flow
Work in Progress

Lennon Ó Náraigh [1], Daniel R. Jansen van Vuuren [1,2], Mohsen Sharifpur [2]

[1]School of Mathematics and Statistics, University College Dublin, Belfield, Dublin 4, Ireland
[1]School of Engineering, University of Pretoria, cnr Lynnwood Road and Roper Street, Hatfield, Pretoria South Africa

December 3, 2019

Introduction

In this talk, we look at two separate problems in the modelling and simulation of both single-phase and multiphase flow involving heat transfer. We bring new methodologies to bear on these problems:

- Theoretical modelling of evaporating sessile droplets (**multiphase**)
- Numerical modelling of heat and momentum transfer in particle-laden channel flows (**single phase**)

Evaporating Sessile Droplets – Context

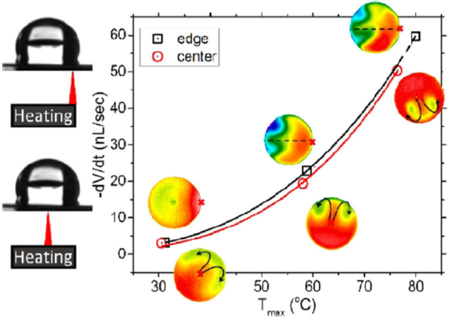


Figure: Askounis et al., Langmuir, 2017.

- Droplet Locally heated at hotspot;
- Evaporation into surrounding atmosphere;
- Marangoni effect induces vortices in droplet.
- Marangoni effect predicted in pure water but was only recently confirmed experimentally (contaminants).

Evaporating Sessile Droplets – Modelling Problem

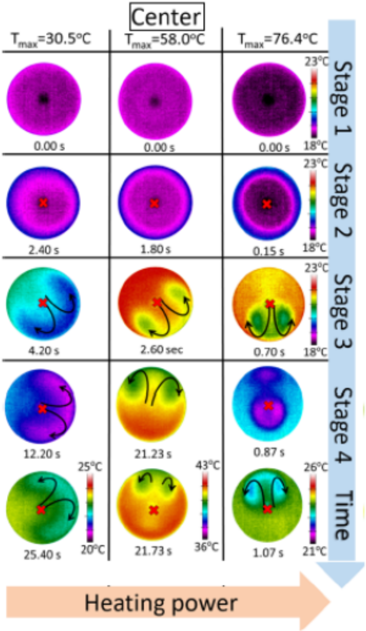


Figure from Askounis et al., Langumuir, 2017:

- Droplet heated at hotspot with laser – constant heat fluix.
- **Symmetric** temperature distribution emerges.
- After some time, convection sets in.
- Convection happens fast, then evaporation n a much slower timescale.
- Vortices move around in the droplet in a dynamic fashion, suggesting non-linear behaviour.

Modelling Assumptions

The aim of the research is to develop a theoretical model for predicting the onset of the convection. As such, the following simplifying assumptions can be made:

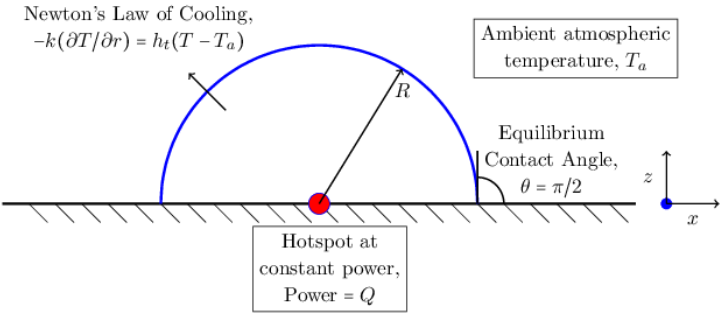
- Convection sets in long before evaporation starts – assume droplet keeps its shape in the model.
- Equilibrium contact angle $\theta \approx 110^\circ$ – treat as hemispherical in the model.
- Idea – develop a temperature distribution for the case without convection – **base state**.
- Treat the onset of convection as a small-amplitude perturbation and develop a **linear stability analysis** of the system.
- Further assumptions are required for the boundary conditions.

Boundary Conditions

- One-sided model – treat only what happens inside liquid phase.
- Gas phase parametrized by Newton's Law of Cooling, and hence, a Robin boundary condition at $r = R$:

$$k(\partial T/\partial r) = h_t(T - T_{\mathrm{a}}),$$

  where T is the droplet temperature, k is the thermal conductivity, h_t is the coefficient in Newton's Law of Cooling, and T_{a} is the ambient gas temperature.
- Neumann / Dirichlet conditions are applied at $z = 0$, as appropriate.



Base State

The base state describes what happens in the absence of flow. It should be the solution to the diffusion equation

$$\frac{\partial T_*}{\partial t} = \nabla^2 T_*, \qquad \text{in the hemisphere,}$$

subject to the appropriate boundary conditions at the substrate. The choice of boundary conditions is crucial.

- Realistic boundary conditions at $z = 0$, e.g. $k(\partial T/\partial z) = f(r)$;
- Here, $f(r)$ is the source function which depends on the laser power (e.g. Gaussian)
- Then, T_* is not radially symmetric but instead depends on both r and z: $T_* = T_*(r, z)$ – agrees with experimental observations.
- We have also investigated a radially symmetric solution with $T_* \to T_*(r)$.

For now we leave the T_* unspecified and outline the linear stability analysis in broad terms.

Fluid Dynamics

Beyond the base state, we introduce the Navier–Stokes equations for viscous incompressible flow:

$$\rho_0 \left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) = -\nabla p + \mu \nabla^2 \boldsymbol{u} - \rho_0 g \left[1 - \alpha (T - T_{\mathrm{a}}) \right] \widehat{\boldsymbol{z}},$$

Fluid Dynamics

Beyond the base state, we introduce the Navier–Stokes equations for viscous incompressible flow:

$$\rho_0 \left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) = -\nabla p + \mu \nabla^2 \boldsymbol{u} - \rho_0 g \left[1 - \alpha (T - T_{\mathrm{a}}) \right] \widehat{\boldsymbol{z}},$$

- Work in Bousinesseq limit, where ρ_0 denotes a constant reference density; also, μ is the viscosity.
- We **locally** use α to denote the coeffcient of thermal expansion.
- Calculations suggest $\mathrm{Ma} \gg \mathrm{Ra}$ (defined later); hence, buoyancy term can be dropped.

Fluid Dynamics

Beyond the base state, we introduce the Navier–Stokes equations for viscous incompressible flow:

$$\rho_0\left(\frac{\partial \boldsymbol{u}}{\partial t}+\boldsymbol{u}\cdot\nabla\boldsymbol{u}\right)=-\nabla p+\mu\nabla^2\boldsymbol{u}-\rho_0 g\left[1-\alpha(T-T_{\mathrm{a}})\right]\widehat{\boldsymbol{z}},$$

- Work in Bousinesseq limit, where ρ_0 denotes a constant reference density; also, μ is the viscosity.
- We **locally** use α to denote the coeffcient of thermal expansion.
- Calculations suggest $\mathrm{Ma}\gg\mathrm{Ra}$ (defined later); hence, buoyancy term can be dropped.

Equations of motion simplify (and supplemented by incompressibility):

$$\begin{aligned}\rho_0\left(\frac{\partial \boldsymbol{u}}{\partial t}+\boldsymbol{u}\cdot\nabla\boldsymbol{u}\right) &= -\nabla p+\mu\nabla^2\boldsymbol{u}-\rho_0 g\widehat{\boldsymbol{z}},\\ \nabla\cdot\boldsymbol{u} &= 0\end{aligned}$$

(we henceforth drop the subscript on the density, for consistency). Finally, introduce advection-diffusion equation for the temperature:

$$\rho C_p\left(\frac{\partial T}{\partial t}+\boldsymbol{u}\cdot\nabla T\right)=k\nabla^2 T.$$

Fluid Dynamics – Boundary Conditions

- No slip: $\boldsymbol{u} = 0$ at $z = 0$.
- No mass flux at interface (evaporation suppressed at short times) – radial velocity condition: $u_r = 0$ at $r = R$.
- Also, Marangoni stress condition at the interface, since the surface tension is a function of temperature:

$$\sigma = \sigma_0 - \gamma \left(T - T_{\mathrm{a}} \right).$$

Fluid Dynamics – Boundary Conditions

- No slip: $\boldsymbol{u} = 0$ at $z = 0$.
- No mass flux at interface (evaporation suppressed at short times) – radial velocity condition: $u_r = 0$ at $r = R$.
- Also, Marangoni stress condition at the interface, since the surface tension is a function of temperature:

$$\sigma = \sigma_0 - \gamma \left(T - T_{\mathrm{a}} \right).$$

Effective **vorticity source**:

$$\begin{aligned} \mu \widehat{\boldsymbol{r}} \cdot \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T \right) \cdot \widehat{\boldsymbol{\theta}} &= -\frac{\gamma}{R} \frac{\partial T}{\partial \theta}, \\ \mu \widehat{\boldsymbol{r}} \cdot \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T \right) \cdot \widehat{\boldsymbol{\varphi}} &= -\frac{\gamma}{R \sin \theta} \frac{\partial T}{\partial \varphi}. \end{aligned}$$

Fluid Dynamics – Boundary Conditions

- No slip: $\boldsymbol{u} = 0$ at $z = 0$.
- No mass flux at interface (evaporation suppressed at short times) – radial velocity condition: $u_r = 0$ at $r = R$.
- Also, Marangoni stress condition at the interface, since the surface tension is a function of temperature:

$$\sigma = \sigma_0 - \gamma \left(T - T_\mathrm{a} \right).$$

Effective **vorticity source**:

$$\begin{aligned} \mu \widehat{\boldsymbol{r}} \cdot \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T \right) \cdot \widehat{\boldsymbol{\theta}} &= -\frac{\gamma}{R} \frac{\partial T}{\partial \theta}, \\ \mu \widehat{\boldsymbol{r}} \cdot \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T \right) \cdot \widehat{\boldsymbol{\varphi}} &= -\frac{\gamma}{R \sin\theta} \frac{\partial T}{\partial \varphi}. \end{aligned}$$

With $u_r = 0$ at $r = R$, these equations simplify:

$$\begin{aligned} \mu \frac{\partial}{\partial r} \left(\frac{u_\theta}{r} \right) \bigg|_{r=R} &= -\frac{\gamma}{R^2} \frac{\partial T}{\partial \theta} \bigg|_{r=R}, \\ \mu \frac{\partial}{\partial r} \left(\frac{u_\varphi}{r} \right) \bigg|_{r=R} &= -\frac{\gamma}{R^2 \sin\theta} \frac{\partial T}{\partial \varphi}. \end{aligned}$$

Linear Stability Analysis and Key Boundary Condition

- We introduce solutions of the Navier–Stokes equations which are a small perturbation around the base state.
- Fluid velocities are assumed to have a small amplitude;
- Temperature distribution given by

$$T = \underbrace{T_*(r,\theta,t)}_{\text{Base State}} + \delta T(r,\theta,\varphi,t).$$

If the model for T_* has a Neumann boundary condition at the substrate, (i.e. $k\partial T_*/\partial z = f(r)$), then δT should also have a Neumann boundary condition at the substrate – a **homogeneous** one, $\partial_z \delta T = 0$.

Linearized Equations of Motion

- In linear stability, the term $\boldsymbol{u}\cdot\nabla\boldsymbol{u}$ is omitted from the equations of motion.
- We assume that T_* varies very slowly, such that $\partial T_*/\partial t$ is ignored.
- By acting repeatedly on the resulting equations with the curl operator, we therefore obtain (following Ha and Lai, Proc. Lond. Roy. Soc. A, 2000):

$$\begin{aligned}\nabla^2\left(\nu\nabla^2-\partial_t\right)(ru_r) &= 0,\\ (\kappa\nabla^2-\partial_t)\delta T &= u_r(\partial T_*/\partial z)+u_\theta(\partial T_*/\partial\theta),\end{aligned}$$

where $\kappa=k/(\rho C_p)$ is the thermal diffusivity of the water.

Linearized Equations of Motion

- In linear stability, the term $\boldsymbol{u}\cdot\nabla\boldsymbol{u}$ is omitted from the equations of motion.
- We assume that T_* varies very slowly, such that $\partial T_*/\partial t$ is ignored.
- By acting repeatedly on the resulting equations with the curl operator, we therefore obtain (following Ha and Lai, Proc. Lond. Roy. Soc. A, 2000):

$$
\begin{aligned}
\nabla^2\left(\nu\nabla^2-\partial_t\right)(ru_r) &= 0,\\
(\kappa\nabla^2-\partial_t)\delta T &= u_r(\partial T_*/\partial z)+u_\theta(\partial T_*/\partial\theta),
\end{aligned}
$$

where $\kappa = k/(\rho C_p)$ is the thermal diffusivity of the water.

Boundary conditions at $r=R$ simplify (Ha and Li):

$$
\begin{aligned}
\frac{\partial^2}{\partial r^2}(ru_r)-\frac{2+\nabla_\Omega^2}{r^2}(ru_r) &= \frac{\gamma}{r}\nabla_\Omega^2\delta T,\\
-k\partial_r\delta T &= h_t\delta T,\\
u_r &= 0.
\end{aligned}
$$

Here, ∇_Ω^2 is the Laplace–Beltrami operator on the sphere.

Linear Stability Analysis

We work at criticality such that $\partial_t = 0$. The aim of the remaining analysis (still to be done) is to solve

$$\begin{aligned} \nabla^4(ru_r) &= 0, \\ \nabla^2 \delta T &= u_r(\partial T_*/\partial z) + u_\theta(\partial T_*/\partial \theta), \end{aligned}$$

subject to the given boundary conditions.

Linear Stability Analysis

We work at criticality such that $\partial_t = 0$. The aim of the remaining analysis (still to be done) is to solve

$$\begin{aligned} \nabla^4(ru_r) &= 0, \\ \nabla^2 \delta T &= u_r(\partial T_*/\partial z) + u_\theta(\partial T_*/\partial \theta), \end{aligned}$$

subject to the given boundary conditions.

- By analogy to a prior study with T_* radially symmetric, we expect this to yield a solvability condition.
- Hence, we expect a consistent solution to these equations exist only for a critical value of γ dependent on the base-state temperature T_*:

$$\mathrm{Ma} = \frac{\gamma R[Q/(2\pi k)]}{\kappa\mu} = \Phi(\langle T_* \rangle),$$

- Here, $\Phi(\langle T_* \rangle)$ is to be determined, and the angle brackets denote averaging over space.

Linear Stability Analysis

We work at criticality such that $\partial_t = 0$. The aim of the remaining analysis (still to be done) is to solve

$$\begin{aligned} \nabla^4(ru_r) &= 0, \\ \nabla^2 \delta T &= u_r(\partial T_*/\partial z) + u_\theta(\partial T_*/\partial \theta), \end{aligned}$$

subject to the given boundary conditions.

- By analogy to a prior study with T_* radially symmetric, we expect this to yield a solvability condition.
- Hence, we expect a consistent solution to these equations exist only for a critical value of γ dependent on the base-state temperature T_*:

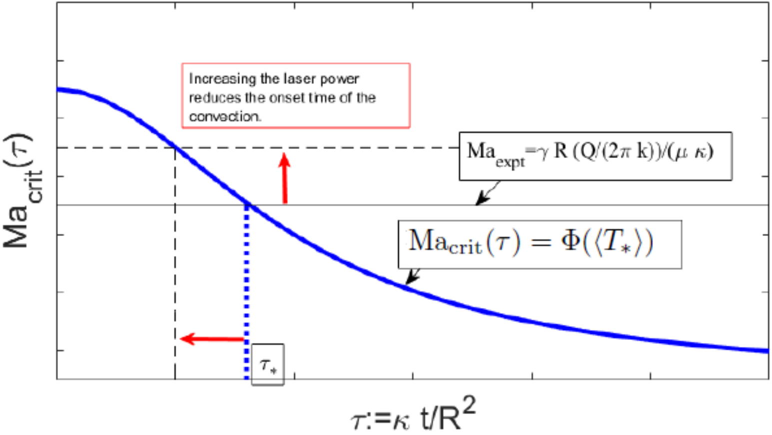
$$\mathrm{Ma} = \frac{\gamma R[Q/(2\pi k)]}{\kappa\mu} = \Phi(\langle T_* \rangle),$$

- Here, $\Phi(\langle T_* \rangle)$ is to be determined, and the angle brackets denote averaging over space.

We expect the time-dependence to enter via $Ma_{\mathrm{crit}} = \Phi(\langle T_* \rangle)$, where T_* depends weakly on time.

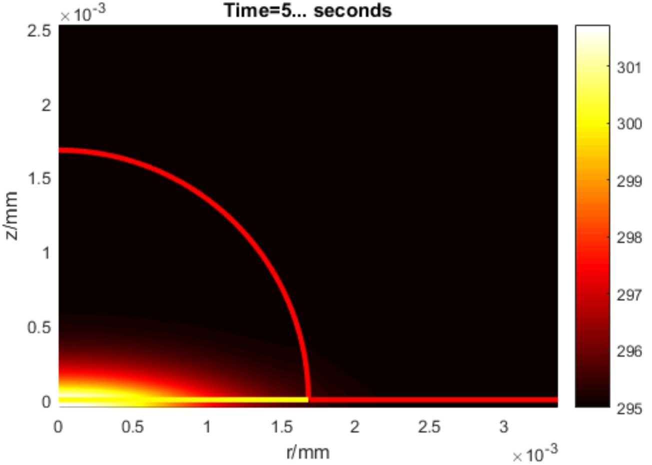
Graphical representation of the criterion for onset of convection

The idea will be to solve the PDEs numerically and obtain $\mathrm{Ma}_{\mathrm{crit}}(\tau) = \Phi(\langle T_* \rangle)$, where the time dependence enters via $\langle T_* \rangle$, and where $\tau = \kappa t / R^2$.



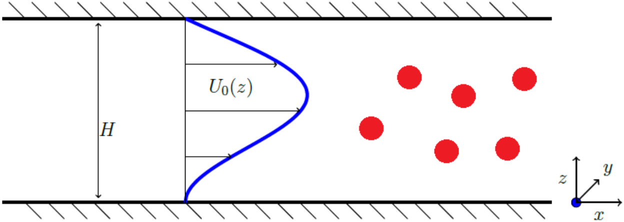
Analytical progress

Analytical progress depends on finding a ‘reasonable’ shape for the base state $T_*(z, r, t)$. Simulation of $\partial_t T_* = \nabla^2 T_*$ will help here, e.g.



Part 2. Particle-laden channel flows

In this part of the talk we outline tentative work in the modelling and simulation of particles in channel flows:

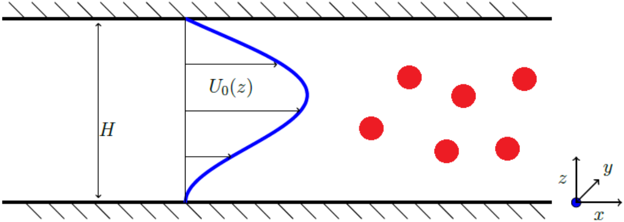


We take an existing parallel flow solver (S-TPLS) and we sequentially add:

- Immersed boundary capabilities to simulate particles (currently only stationary particles);
- Advection-diffusion equation (with immersed boundary method) to model heat transfer

Part 2. Particle-laden channel flows

In this part of the talk we outline tentative work in the modelling and simulation of particles in channel flows:



We take an existing parallel flow solver (S-TPLS) and we sequentially add:

- Immersed boundary capabilities to simulate particles (currently only stationary particles);
- Advection-diffusion equation (with immersed boundary method) to model heat transfer

We use ANSYS-Fluent for validation purposes; **using S-TPLS for modelling purposes** has advantages (e.g. scalability in future high-resolution studies).

S-TPLS – overview

- S-TPLS is a stripped-down version of an in-house two-phase solver; S-TPLS is single-phase.
- S-TPLS solves the incompressible Navier–Stokes equation in a channel, using a finite-volume spatial discretization.

S-TPLS – overview

- S-TPLS is a stripped-down version of an in-house two-phase solver; S-TPLS is single-phase.
- S-TPLS solves the incompressible Navier–Stokes equation in a channel, using a finite-volume spatial discretization.
- Marker-and-cell spatial discretization: pressures and other scalar quantities at cell centres, velocities at cell faces.
- Finite-volumes, with flux-conservative differencing for the momentum equation.

S-TPLS – overview

- S-TPLS is a stripped-down version of an in-house two-phase solver; S-TPLS is single-phase.
- S-TPLS solves the incompressible Navier–Stokes equation in a channel, using a finite-volume spatial discretization.
- Marker-and-cell spatial discretization: pressures and other scalar quantities at cell centres, velocities at cell faces.
- Finite-volumes, with flux-conservative differencing for the momentum equation.
- Momentum step: centred differences for the convective derivative, Crank–Nicolson treatment for the diffusion, third-order Adams–Bashforth for the time evolution.
- Projection method: Momenta are updated first, followed by a correction step involving a pressure update, thereby enforcing incompressibility.

S-TPLS – overview

- S-TPLS is a stripped-down version of an in-house two-phase solver; S-TPLS is single-phase.
- S-TPLS solves the incompressible Navier–Stokes equation in a channel, using a finite-volume spatial discretization.
- Marker-and-cell spatial discretization: pressures and other scalar quantities at cell centres, velocities at cell faces.
- Finite-volumes, with flux-conservative differencing for the momentum equation.
- Momentum step: centred differences for the convective derivative, Crank–Nicolson treatment for the diffusion, third-order Adams–Bashforth for the time evolution.
- Projection method: Momenta are updated first, followed by a correction step involving a pressure update, thereby enforcing incompressibility.
- Code is written in Fortran90 and parallelized using MPI; parallelization scheme takes account of problem geometry (2D domain decomposition)

Immersed boundary method

We use the method of Kajishima et al. (2001) to introduce the solid phase: a solid-body volume fraction is α is introduced, such that

$$\alpha(\boldsymbol{x}) = \begin{cases} 1, \text{if } \boldsymbol{x} \text{ is in the solid phase,} \\ 0, \text{if } \boldsymbol{x} \text{ is in the fluid phase,} \end{cases}$$

note change in use of α !

with $\alpha(\boldsymbol{x})$ transitioning smoothly between the two extreme values.

Immersed boundary method

We use the method of Kajishima et al. (2001) to introduce the solid phase: a solid-body volume fraction is α is introduced, such that

$$\alpha(\boldsymbol{x}) = \begin{cases} 1, \text{if } \boldsymbol{x} \text{ is in the solid phase,} \\ 0, \text{if } \boldsymbol{x} \text{ is in the fluid phase,} \end{cases}$$

note change in use of α !

with $\alpha(\boldsymbol{x})$ transitioning smoothly between the two extreme values.
At the end of the pressure-correction step, the updated velocity is $\mathbf{u}^{n+1}$; this updated velocity is modified further: to enforce $\mathbf{u} = 0$ in the solid phase:

$$\mathbf{u}^{\text{modified}} = \mathbf{u}^{n+1} - \alpha \mathbf{u}^{n+1}$$

This is a simple and robust method and gives good results for **stationary** particles.

Immersed boundary method

We use the method of Kajishima et al. (2001) to introduce the solid phase: a solid-body volume fraction is α is introduced, such that

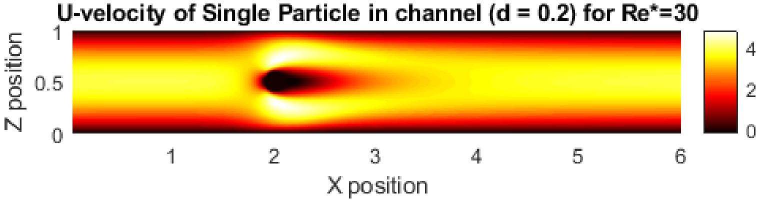
$$\alpha(\boldsymbol{x}) = \begin{cases} 1, \text{if } \boldsymbol{x} \text{ is in the solid phase,} \\ 0, \text{if } \boldsymbol{x} \text{ is in the fluid phase,} \end{cases}$$

note change in use of α !

with $\alpha(\boldsymbol{x})$ transitioning smoothly between the two extreme values.
At the end of the pressure-correction step, the updated velocity is $\mathbf{u}^{n+1}$; this updated velocity is modified further: to enforce $\mathbf{u} = 0$ in the solid phase:

$$\mathbf{u}^{\text{modified}} = \mathbf{u}^{n+1} - \alpha \mathbf{u}^{n+1}$$

This is a simple and robust method and gives good results for **stationary** particles.



Validation I

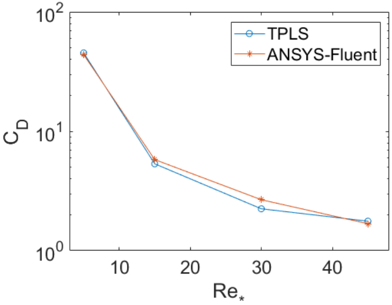
We validate the method by computing the total drag past the cylinder, as a function of Reynolds number Re_*.

The drag coefficient C_D is computed from:

$$C_D = \frac{1}{\rho U^2} \int_\Omega (-p n_x) |\nabla \alpha| \mathrm{d}^2 x$$
$$+ \frac{1}{\rho U^2} \int_\Omega (\tau_{xx} n_x + \tau_{xz} n_z) |\nabla \alpha| \mathrm{d}^2 x,$$

where

$$\tau_{ij} = \frac{1}{\mathrm{Re}_*} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right).$$



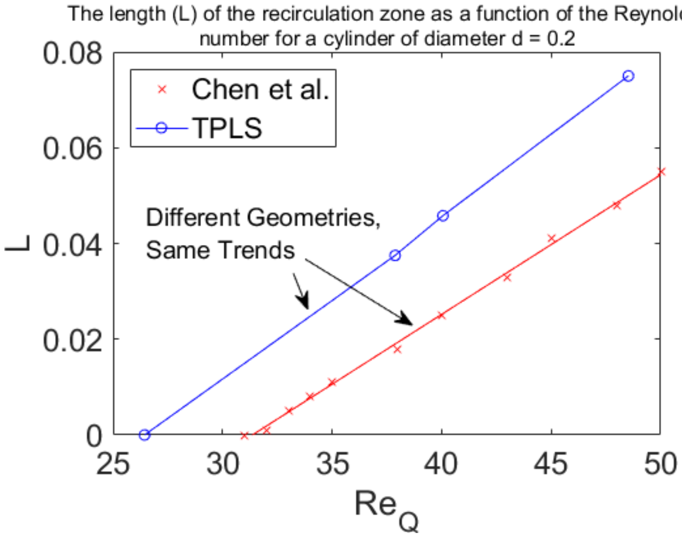
Here, n_x and n_z are obtainable from the solid volume fraction:

$$(n_x, n_z) = \nabla\alpha / |\nabla \alpha| \qquad |\nabla \alpha| \text{ proportional to delta function;}$$

also, U is the mean inlet velocity.

Validation II

We also look at the critical Reynolds number for the onset of recirculation in the cylinder wake (classical problem; depends closely on inlet boundary conditions).



With Advection-Diffusion

We also look at modelling heat transfer by adding the advection-diffusion equation to S-TPLS:

$$\frac{\partial T}{\partial t} + \boldsymbol{u} \cdot \nabla T = \frac{1}{\mathrm{Pe}} \nabla^2 T.$$

- Periodic boundary conditions in x-direction.
- Temperature gradient in wall-normal direction.
- Numerical solution via Crank–Nicolson method.
- Addition of particles: once T^{n+1} is obtained via Crank-Nicolson, temperature is modified to account for the fixed temperature in the particles:

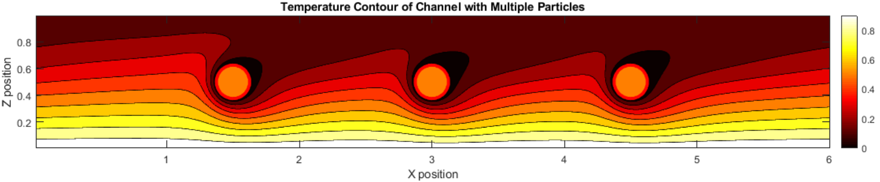
$$T^{\mathsf{modified}} = T^{n+1} + \alpha \left(T_{\mathsf{particle}} - T^{n+1} \right).$$

- Quantify enhancement to heat transfer from particles via Nusselt number:

$$\mathrm{Nu} = \frac{L_x^{-1} \int_\Omega (1-\alpha) \left(wT - \mathrm{Pe}^{-1} \frac{\partial T}{\partial z} \right) \mathrm{d}^2 x}{\mathrm{Pe}^{-1} (T_{\mathsf{bottom}} - T_{\mathrm{top}})}.$$

Sample Results

- Quick check: S-TPLS preserves the constant stratification $T = T_{\text{bottom}} + (T_{\text{top}} - T_{\text{bottom}})(z/L_z)$ in the absence of particles.
- Currently no buoyancy term in momentum equation; this will be added soon.
- We have also looked at one particle, with normalized temperature values $T_{\text{bottom}} = 1$, $T_{\text{top}} = 0.1$, and $T_{\text{particle}} = 0$.
- We are also looking at flow / temperature distributions past arrays of particles.



Conclusions

In **Part 1** we have:

- Looked at the problem of the onset of Marangoni convection in a locally heated sessile droplet.
- Formulated the linear stability analysis up to the point where the base state needs to be specified in concrete terms.
- Outlined how this approach can predict the critical **time** for the onset of Marangoni convection.

Conclusions

In **Part 1** we have:

- Looked at the problem of the onset of Marangoni convection in a locally heated sessile droplet.
- Formulated the linear stability analysis up to the point where the base state needs to be specified in concrete terms.
- Outlined how this approach can predict the critical **time** for the onset of Marangoni convection.

In **Part 2** we have:

- Added solid bodies to the S-TPLS highly parallelized single-phase channel-flow solver.
- Added a temperature equation to model heat transfer
- Outlined how enhancement to heat transfer can be quantified via the **Nusselt number**
- Invite suggestions for which systems to look at next.