

Topics in the modelling and simulation of two-phase flows

Lennon Ó Náraigh ¹, Prashant Valluri ², Iain Bethune ³,
David Scott ³, Peter Spelt ⁴, Selma Shun ¹, Aurore Naso ⁴

¹ School of Mathematical Sciences and CASL, University College Dublin

² Institute for Materials and Processes, University of Edinburgh

³ Edinburgh Parallel Computing Centre, University of Edinburgh

⁴ LMFA, Ecole Centrale de Lyon

18th February 2016

Overview

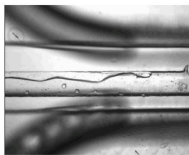
- Motivation
- TPLS computational framework
- Results for two-phase interfacial instabilities
- Simple numerical model for global modes
- Exotic application – phase separation in binary liquids

Context

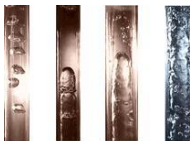
Two-phase stratified flow is ubiquitous in nature.



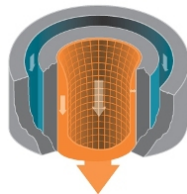
(a) Kelvin-Helmholtz instability



(b) Stratified flow in pipelines



(c) Slug flow

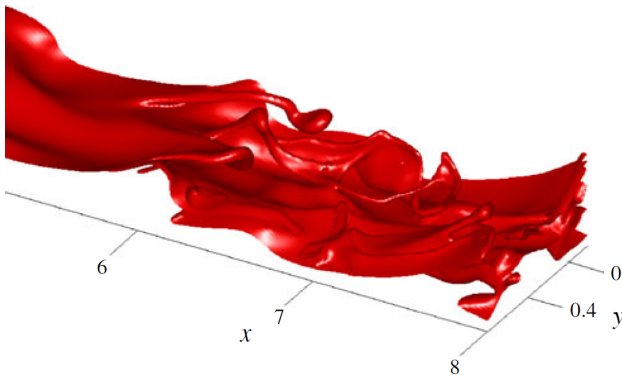


(d) Falling-film reactors

- Mathematically, and computationally, a tough problem – turbulence, extreme nonlinearity, topological change in interfaces, a range of instabilities that need to be captured.
- Even the laminar regime is tough - current focus of the research.

The numerical challenge

- Flows involving many length- and time-scales
- Flows with sharp changes in interfacial topologies
- Transient three-dimensional simulations required over long periods of time, requiring **scalable** codes run at very high **resolutions**.



Existing methodologies

- Existing interface-capturing methods: Levelset, Volume of Fluid, Particles, Diffuse Interface Method

Existing methodologies

- Existing interface-capturing methods: Levelset, Volume of Fluid, Particles, Diffuse Interface Method
- Existing implementations: Open-source (e.g. Gerris, etc.), Commercial (CFX, etc.), in-house solvers.

Existing methodologies

- Existing interface-capturing methods: Levelset, Volume of Fluid, Particles, Diffuse Interface Method
- Existing implementations: Open-source (e.g. Gerris, etc.), Commercial (CFX, etc.), in-house solvers.
- Some drawbacks (not respectively): Black-box approach, validation uncertain, artificial diffusion. Key drawbacks: resolution constraints, scalability.

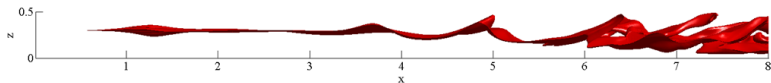
Existing methodologies

- Existing interface-capturing methods: Levelset, Volume of Fluid, Particles, Diffuse Interface Method
- Existing implementations: Open-source (e.g. Gerris, etc.), Commercial (CFX, etc.), in-house solvers.
- Some drawbacks (not respectively): Black-box approach, validation uncertain, artificial diffusion. Key drawbacks: resolution constraints, scalability.
- TPLS addresses these issues, in particular **resolution and scalability**.

Existing methodologies

- Existing interface-capturing methods: Levelset, Volume of Fluid, Particles, Diffuse Interface Method
- Existing implementations: Open-source (e.g. Gerris, etc.), Commercial (CFX, etc.), in-house solvers.
- Some drawbacks (not respectively): Black-box approach, validation uncertain, artificial diffusion. Key drawbacks: resolution constraints, scalability.
- TPLS addresses these issues, in particular **resolution and scalability**.
- Not a silver bullet – levelset methods – tradeoff between capturing interfacial topology with great fidelity, and mass loss. But mass loss minimized at high resolution.

TPLS



Equations of motion

Numerical solution of two-phase Navier–Stokes equations with interface capturing:

$$\rho(\phi) \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \frac{1}{Re} \nabla \cdot \left[\mu(\phi) \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right) \right] + \mathbf{f}_{\text{st}}(\phi) + \rho(\phi) \mathbf{g},$$

where $\nabla \cdot \mathbf{u} = 0$ and ϕ is the interface-capturing field:

Equations of motion

Numerical solution of two-phase Navier–Stokes equations with interface capturing:

$$\rho(\phi) \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \frac{1}{Re} \nabla \cdot \left[\mu(\phi) \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right) \right] + \mathbf{f}_{\text{st}}(\phi) + \rho(\phi) \mathbf{g},$$

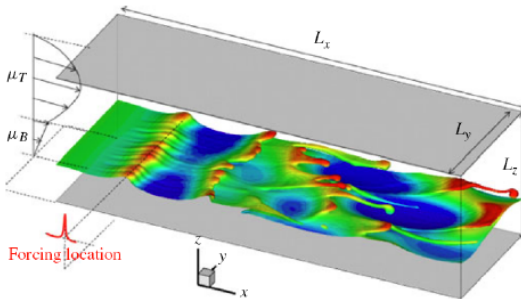
where $\nabla \cdot \mathbf{u} = 0$ and ϕ is the interface-capturing field:

Levelset method:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0, \quad \mathbf{f}_{\text{st}} = \delta_\epsilon(\phi) \frac{1}{We} \hat{\mathbf{n}} \nabla \cdot \hat{\mathbf{n}}, \quad \hat{\mathbf{n}} = \frac{\nabla \phi}{|\nabla \phi|}.$$

Problem geometry and configuration

- Simple channel geometry: periodic OR inlet/outlet conditions at $x = 0$, $x = L_x$; walls (no slip) at $z = 0$, $z = L_z$.
- Basic version involves hydrodynamics only. TPLS with physics available, e.g. evaporating droplets, contact-line dynamics, mass transfer.



Numerical discretization schemes

- Marker-and-cell discretization: pressures, densities, viscosities, and ϕ at cell centres, velocities at cell faces.

Numerical discretization schemes

- Marker-and-cell discretization: pressures, densities, viscosities, and ϕ at cell centres, velocities at cell faces.
- Finite-volumes, with flux-conservative differencing for the momentum equation.

Numerical discretization schemes

- Marker-and-cell discretization: pressures, densities, viscosities, and ϕ 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–Nicholson treatment for the diffusion, third-order Adams–Bashforth for the time evolution.

Numerical discretization schemes

- Marker-and-cell discretization: pressures, densities, viscosities, and ϕ 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–Nicholson 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.

Numerical discretization schemes

- Marker-and-cell discretization: pressures, densities, viscosities, and ϕ 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–Nicholson 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.
- The levelset function $\phi(x, y, z, t)$ is carried with the flow (3rd-order WENO) but is corrected at each timestep (‘redistancing’).

Parallel computing

- Typical runs involve up to 10 million gridpoints, meaning that large-scale parallel simulation is unavoidable (larger runs (up to 30 million gridpoints) have been performed).

Parallel computing

- Typical runs involve up to 10 million gridpoints, meaning that large-scale parallel simulation is unavoidable (larger runs (up to 30 million gridpoints) have been performed).
- Code is parallelized using hybrid MPI technology; parallelization scheme takes account of problem geometry (2D domain decomposition)

Parallel computing

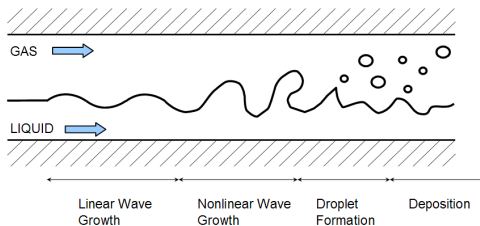
- Typical runs involve up to 10 million gridpoints, meaning that large-scale parallel simulation is unavoidable (larger runs (up to 30 million gridpoints) have been performed).
- Code is parallelized using hybrid MPI technology; parallelization scheme takes account of problem geometry (2D domain decomposition)
- Code has been optimized as part of joint work with EPCC (SSI and DCSE projects): synchronization barriers have been systematically removed, pressure-correction step has been optimized with PETSC linear-algebra library (TPLS 1.0). Customizable TPLS with Parallel I/O with NetCDF data storage (TPLS 2.0)
- Current implementation of pressure solver uses GMRES with block-Jacobi preconditioner. Original crude SOR version still available (and robust!).

Parallel computing

- Typical runs involve up to 10 million gridpoints, meaning that large-scale parallel simulation is unavoidable (larger runs (up to 30 million gridpoints) have been performed).
- Code is parallelized using hybrid MPI technology; parallelization scheme takes account of problem geometry (2D domain decomposition)
- Code has been optimized as part of joint work with EPCC (SSI and DCSE projects): synchronization barriers have been systematically removed, pressure-correction step has been optimized with PETSC linear-algebra library (TPLS 1.0). Customizable TPLS with Parallel I/O with NetCDF data storage (TPLS 2.0)
- Current implementation of pressure solver uses GMRES with block-Jacobi preconditioner. Original crude SOR version still available (and robust!).

Strict benchmarks for code's accuracy

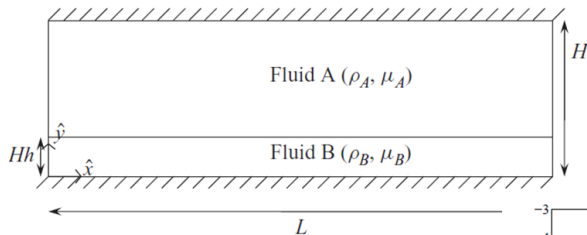
- Introduce a tiny sinusoidal perturbation at the interface.
- Produces pressure and velocity fluctuations that satisfy linear equations of motion.
- Linearized equations of motion solved via eigenvalue analysis (independent, quasi-analytical).
- Gives growth rate and wave speed of wave-like fluctuations.



Focus on finding agreement between OS analysis and wave growth in the code.

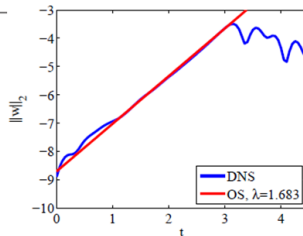
Orr–Sommerfeld analysis – Results

Stratified co-flow test case ($h=0.3$)

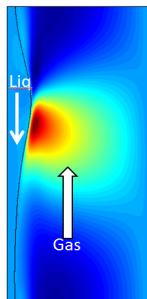


$$(Re, r, m, \mathcal{S}) = (100, 1, 30, 0.01)$$

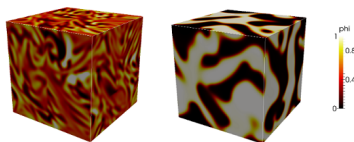
- $L \times W \times H = (3 \times 1 \times 1)$
- 12 million grid points
- 1024 processors, 12 h



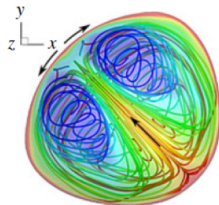
Applications of TPLS



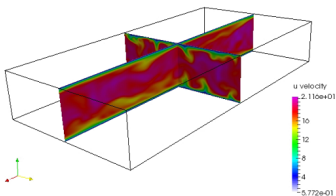
Countercurrent gas-liquid flows



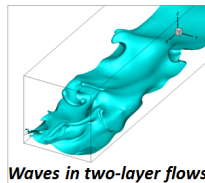
Phase separation in binary liquids



Evaporation of non-spherical droplets



Large-eddy simulation



Waves in two-layer flows

Take-home message

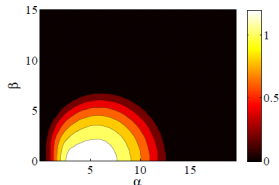
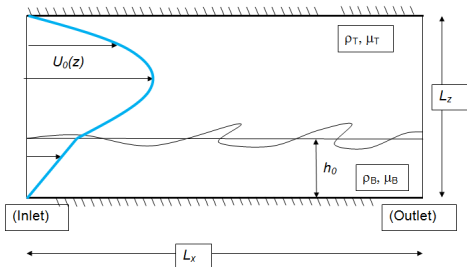
- DNS are one of the most important tools to examine multiphase flows
- A wide range of 3D two-phase phenomena can be modelled through full numerical simulations using TPLS.
- Next aim is to merge all versions / models to have a user-customisable multiphysics version of TPLS
- New applications / collaborations always welcome
- TPLS is available as opensource solution for further applications:

`http://sourceforge.net/projects/tpls/`

*Some results for
liquid-liquid flows*

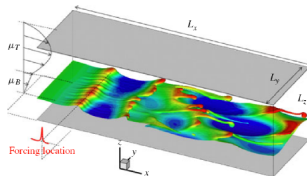


Where do 3D waves in parallel flows come from?



$$r = 1, \mathcal{S} = 0.1$$

Linear instability of 2D parallel flow is dominated by 2D waves.
So how do 3D structures form?



We want to keep an open mind and examine all possibilities.

- Direct route for supercritical cases – wherein linear theory predicts 2D and 3D waves are present in more-or-less equal strengths.

We want to keep an open mind and examine all possibilities.

- Direct route for supercritical cases – wherein linear theory predicts 2D and 3D waves are present in more-or-less equal strengths.
- For subcritical cases – possibility of linear transient growth.

We want to keep an open mind and examine all possibilities.

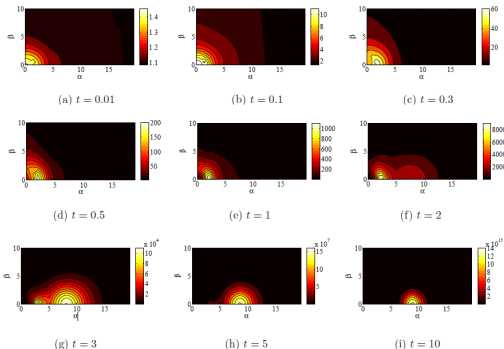
- Direct route for supercritical cases – wherein linear theory predicts 2D and 3D waves are present in more-or-less equal strengths.
- For subcritical cases – possibility of linear transient growth.
- For subcritical cases again – investigate possibility of nonlinear route.

Transient growth is not directly relevant.

maximum amplification factor:

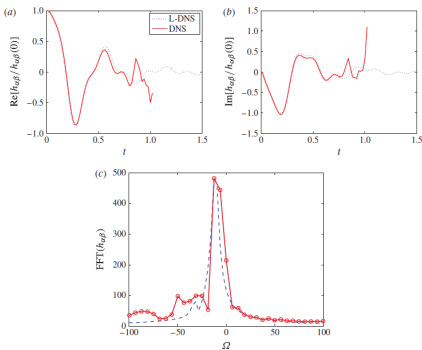
$$G_{\alpha\beta}(t) = \sup \| (w_{\alpha\beta}(z, t), \omega_{z, \alpha\beta}(z, t)) \|_E,$$

where $\| \cdot \|_E$ denotes the energy norm



Snapshots of the maximum amplification factor $G_{\alpha\beta}(t)$ for the parameter values $(m, Re, \mathcal{S}) = (30, 100, 0.01)$

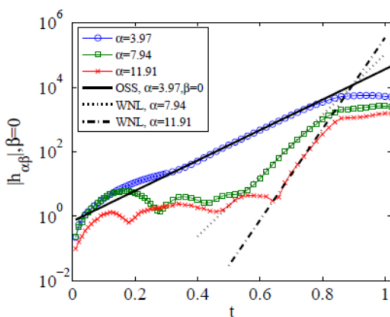
$$\eta(x, y, t = 0) = h_0 + \frac{1}{16} A_0 \sum_{i=0}^3 \sum_{j=0}^3 \cos \left(i \frac{2\pi}{L_x} (1 + \delta)x + j \frac{2\pi}{L_x} (1 + \delta)y + \varphi_{ij} \right)$$



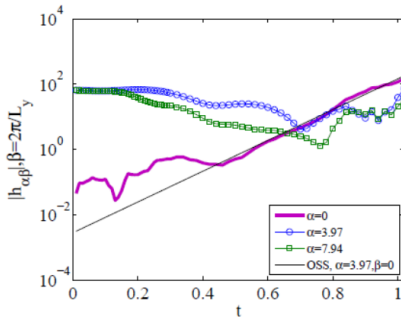
Dynamics of the spanwise Fourier mode $\alpha = (\alpha_0, \beta_0)$.
 (a) Real part of $h_{\alpha}(t)/h_{\alpha}(0)$ with a comparison between the full DNS and the linearized DNS. (b) Imaginary part of the same. (c) Spectral analysis (frequency space) based on the linearized DNS (broken line) and the full nonlinear DNS (solid line, with dots); spectrum taken between $t=0$ and $t=1.02$; FFT = fast Fourier transform. The two most prominent peaks in the L-DNS in fact coincide with the OSS modal frequencies.

Weakly nonlinear route below ‘criticality’

Streamwise waves – Large temporal growth, Spanwise waves – No temporal growth rate



- Streamwise overtones are enslaved to the streamwise dominant mode



- Purely spanwise mode enslaved to the dominant streamwise mode

Periodic boundary conditions, $(Re, m, r, \mathcal{S}) = (300, 30, 1, 0.3)$.

Spanwise modes

- Spanwise mode $(\alpha, \beta) = (0, 2\pi/L_y)$ appears not to grow transiently but rather is slaved to streamwise fundamental.

Spanwise modes

- Spanwise mode $(\alpha, \beta) = (0, 2\pi/L_y)$ appears not to grow transiently but rather is slaved to streamwise fundamental.
- Other info not shown - spanwise mode has extremely long temporal frequency: it does not oscillate at the streamwise fundamental frequency.

Spanwise modes

- Spanwise mode $(\alpha, \beta) = (0, 2\pi/L_y)$ appears not to grow transiently but rather is slaved to streamwise fundamental.
- Other info not shown - spanwise mode has extremely long temporal frequency: it does not oscillate at the streamwise fundamental frequency.
- Suggests weakly nonlinear theory to explain slaving but with some modifications to explain frequency mismatch:

Spanwise modes

- Spanwise mode $(\alpha, \beta) = (0, 2\pi/L_y)$ appears not to grow transiently but rather is slaved to streamwise fundamental.
- Other info not shown - spanwise mode has extremely long temporal frequency: it does not oscillate at the streamwise fundamental frequency.
- Suggests weakly nonlinear theory to explain slaving but with some modifications to explain frequency mismatch:
 - We have obtained a very weakly nonlinear equation for the spanwise-only amplitude, with an effective forcing on the right-hand side (due to all other relevant modes).
 - Selected forcing frequency is got from the **resolvent norm** of the OSS equations.

Strong nonlinear regime

- Study response of system to impulsive disturbance in the limit as the disturbance amplitude becomes finite.
- Neumann BCs at outlet, fixed conditions at inlet.
- Self-sustained oscillations are found - not only do disturbances grow as they are propagated downstream, but the disturbance grows *in situ*, thereby continuously drawing energy into the disturbances at the impulse location.
- Hence, nonlinear absolute instability.
- Absolute instability + nonlinearity \implies **global mode**.

Global modes I

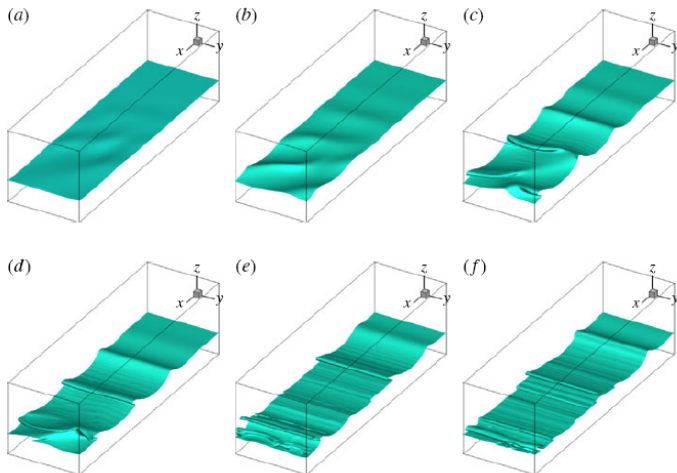


FIGURE 18. (Colour online) Parameter study for case CH2: impulsively forced case with impulsive forcing located initially at $x=L_0=1$: (a) $t=2.0$; (b) $t=3.0$; (c) $t=4.0$; (d) $t=4.5$; (e) $t=8.0$; (f) $t=10.0$.

Global modes II

- A global mode is a nonlinear self-sustained oscillation.
- Two kinds - pushed and pulled (pulled inherits all properties from linear theory).
- DNS confirms global mode is ‘pulled’.

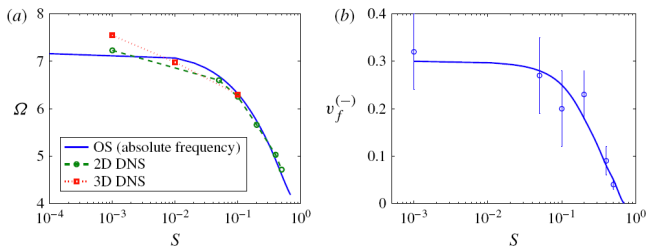
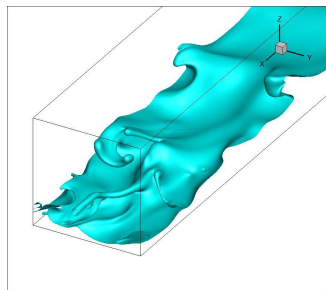


FIGURE 20. (Colour online) (a) Measured global-mode frequency from DNS (both two- and three-dimensional), compared to the linear absolute (two-dimensional) frequency. The frequency is extracted from the time series of the averaged velocity $\int_0^1 dx \int_0^1 dy w(x, y, z, t)$ at location $x = L_0 = 1$. (b) Velocity of upstream-propagating front: comparison between linear theory and two-dimensional DNS. Note that the upstream-propagating front is eventually frozen-in, owing to the Dirichlet boundary condition at the inlet.

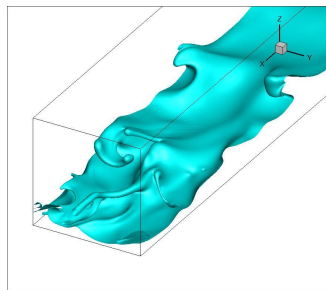
Strong nonlinear regime – conclusions

- Linear theory is embarrassingly good at predicting nonlinear properties because global mode is pulled.



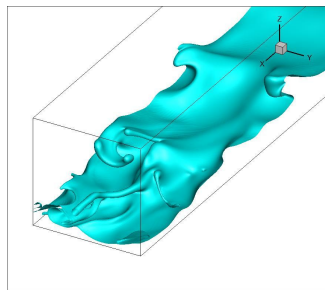
Strong nonlinear regime – conclusions

- Linear theory is embarrassingly good at predicting nonlinear properties because global mode is pulled.
- But global-mode theory not sufficient as the nonlinearity is not saturating.

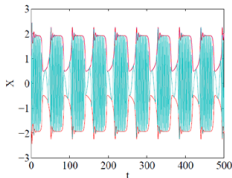


Strong nonlinear regime – conclusions

- Linear theory is embarrassingly good at predicting nonlinear properties because global mode is pulled.
- But global-mode theory not sufficient as the nonlinearity is not saturating.
- Indeed, large-amplitude waves are followed by wave overturning and hence ligament and droplet formation.
- Also, passage from small-amplitude 3D waves to overturning and ligaments needs to be elucidated – **kinematic description**



Global modes – toy models



Motivation

- Obtain some theoretical understanding of self-sustained oscillations
- Understand the connection (if any) to transient growth
- To investigate the possibility that fluid phenomena (transient growth, subcritical transition, self-sustained nonlinear behaviour) can be realised in an optical analogy, thereby opening up the possibility of having a **fluids lab on an optical bench**.

Two-level system – linear theory I

Two-level system:

$$i \frac{\partial u}{\partial t} = \mathcal{H}u + i(\mu_0 \mathbb{I} + \mathcal{G})u, \quad u \in \mathbb{C}^2,$$

where

$$\mathcal{H} = \begin{pmatrix} E_0 & A \\ A & E_0 \end{pmatrix}, \quad \mathcal{G} = \text{diag}(-g_1, -g_2).$$

Note that $[\mathcal{H}, \mathcal{G}] \neq 0$ implies that the operator

$$\mathcal{L} = \mathcal{H} + i(\mu_0 \mathbb{I} + \mathcal{G})$$

is **non-normal**, with $[\mathcal{L}, \mathcal{L}^\dagger] \propto g_2 - g_1$.

Two-level system – linear theory II

Eigenvalues: let $u(t) = u_0 e^{-i\omega t}$, to give

$$\Omega_r = E_0 \pm \sqrt{4A^2 - (g_1 - g_2)^2}, \quad \Omega_i = \mu_0 - \frac{1}{2}(g_1 + g_2),$$

$$4A^2 > (g_1 - g_2)^2, \quad \text{Case 1,}$$

$$\Omega_r = E_0, \quad \Omega_i = \mu_0 - \frac{1}{2}(g_1 + g_2) \pm \sqrt{(g_1 - g_2)^2 - 4A^2}, \quad \text{Case 2.}$$

We work in **Case 2** (crossover is called the diabolic point).

Two-level system – linear theory II

Eigenvalues: let $u(t) = u_0 e^{-i\omega t}$, to give

$$\Omega_r = E_0 \pm \sqrt{4A^2 - (g_1 - g_2)^2}, \quad \Omega_i = \mu_0 - \frac{1}{2}(g_1 + g_2),$$

$$4A^2 > (g_1 - g_2)^2, \quad \text{Case 1,}$$

$$\Omega_r = E_0, \quad \Omega_i = \mu_0 - \frac{1}{2}(g_1 + g_2) \pm \sqrt{(g_1 - g_2)^2 - 4A^2}, \quad \text{Case 2.}$$

We work in **Case 2** (crossover is called the diabolic point).

We use

$$\frac{1}{2} \frac{d}{dt} \|u\|_2^2 \leq [\mu_0 - \min(g_1, g_2)] \|u\|_2^2$$

to identify subcritical parameter values for the forcing μ_0 where transient growth is possible:

$$\min(g_1, g_2) < \mu_0 < \frac{1}{2}(g_1 + g_2) - \sqrt{(g_1 - g_2)^2 - 4A^2}.$$

Introduction of nonlinear terms

- Nonlinear two-level system:

$$i \frac{\partial u}{\partial t} = \mathcal{L}u + a \begin{pmatrix} |u_1|^2 & 0 \\ 0 & |u_2|^2 \end{pmatrix} u.$$

Introduction of nonlinear terms

- Nonlinear two-level system:

$$i \frac{\partial u}{\partial t} = \mathcal{L}u + a \begin{pmatrix} |u_1|^2 & 0 \\ 0 & |u_2|^2 \end{pmatrix} u.$$

- We search for a **self-sustained oscillatory solution**:

$$u = \operatorname{Re} e^{i\Omega t} u_0, \quad \|u_0\|_2^2 = 1, \quad \Omega \in \mathbb{R}.$$

Introduction of nonlinear terms

- Nonlinear two-level system:

$$i \frac{\partial u}{\partial t} = \mathcal{L}u + a \begin{pmatrix} |u_1|^2 & 0 \\ 0 & |u_2|^2 \end{pmatrix} u.$$

- We search for a **self-sustained oscillatory solution**:

$$u = R e^{i\Omega t} u_0, \quad \|u_0\|_2^2 = 1, \quad \Omega \in \mathbb{R}.$$

- Such a solution can be found for $g_2 < \mu_0 < g_2$: we have $\Omega = E_0 + aR^2$, where R has the special value

$$R^2 = \frac{g_1 - g_2}{a} \sqrt{\frac{1}{X^2} - 1}, \quad X^2 = -\frac{(\mu_0 - g_1)(\mu_0 - g_2)}{A^2}.$$

Introduction of nonlinear terms

- Nonlinear two-level system:

$$i \frac{\partial u}{\partial t} = \mathcal{L}u + a \begin{pmatrix} |u_1|^2 & 0 \\ 0 & |u_2|^2 \end{pmatrix} u.$$

- We search for a **self-sustained oscillatory solution**:

$$u = R e^{i\Omega t} u_0, \quad \|u_0\|_2^2 = 1, \quad \Omega \in \mathbb{R}.$$

- Such a solution can be found for $g_2 < \mu_0 < g_2$: we have $\Omega = E_0 + aR^2$, where R has the special value

$$R^2 = \frac{g_1 - g_2}{a} \sqrt{\frac{1}{X^2} - 1}, \quad X^2 = -\frac{(\mu_0 - g_1)(\mu_0 - g_2)}{A^2}.$$

- Recall, the linearized problem was non-normal for $g_1 \neq g_2$. The same condition implies the existence of the non-trivial nonlinear solution!

Introduction of nonlinear terms

- Nonlinear two-level system:

$$i \frac{\partial u}{\partial t} = \mathcal{L}u + a \begin{pmatrix} |u_1|^2 & 0 \\ 0 & |u_2|^2 \end{pmatrix} u.$$

- We search for a **self-sustained oscillatory solution**:

$$u = R e^{i\Omega t} u_0, \quad \|u_0\|_2^2 = 1, \quad \Omega \in \mathbb{R}.$$

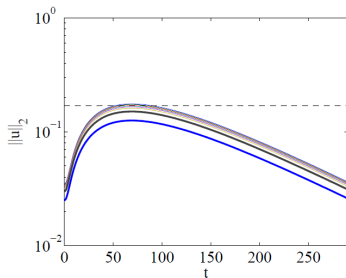
- Such a solution can be found for $g_2 < \mu_0 < g_2$: we have $\Omega = E_0 + aR^2$, where R has the special value

$$R^2 = \frac{g_1 - g_2}{a} \sqrt{\frac{1}{X^2} - 1}, \quad X^2 = -\frac{(\mu_0 - g_1)(\mu_0 - g_2)}{A^2}.$$

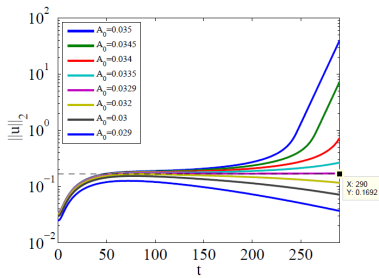
- Recall, the linearized problem was non-normal for $g_1 \neq g_2$. The same condition implies the existence of the non-trivial nonlinear solution!
- Floquet analysis reveals that the nonlinear oscillation is always unstable to a secondary instability (exact result).

Numerical simulation

Numerical simulation with 8th-order accurate Runge-Kutta scheme, with initial condition $u(t=0) = (A_0/\sqrt{2})(i, 1)^T$.



(a)



(b)

FIG. 2. Solutions of (a) the non-Hermitian linear Schrödinger equation; (b) the non-Hermitian *nonlinear* Schrödinger equation. The initial data are the parameters are the same in (a) and (b).

Inhomogeneous CGL equation – linear theory I

A second toy model is the CGL equation:

$$i \frac{\partial u}{\partial t} = \mathcal{H}u(x, t) + i(\mu_0 + \mathcal{G})u(x, t), \quad (x, t) \in \mathbb{R} \times (0, \infty), \quad \mu_0 \in \mathbb{R}$$

with

$$u(x, t = 0) = u_0(x), \quad \lim_{|x| \rightarrow \infty} |u(x, t)| = 0, \quad t \text{ finite,}$$

where

$$\mathcal{H} = -iU \frac{\partial}{\partial x} - \gamma_i \frac{\partial^2}{\partial x^2} + \frac{1}{2} \mu_{2i} x^2, \quad \mathcal{G} = \gamma_r \frac{\partial^2}{\partial x^2} - \frac{1}{2} \mu_{2r} x^2.$$

Again, for $\mathcal{L} = \mathcal{H} + i(\mu_0 + \mathcal{G})$ we have a non-normal evolutionary equation, with $[\mathcal{L}, \mathcal{L}^\dagger] \neq 0$.

Introduction of nonlinear terms I

- Linear theory well understood - eigenvalues/eigenvectors, transient growth.
- We again introduce a nonlinear term in a natural way:

$$i \frac{\partial u}{\partial t} = \mathcal{L}u + a|u|^2 u;$$

- We search for a self-sustained oscillatory solution:

$$u(x, t) = \text{Re} e^{i\Omega t} u_0(x), \quad \|u_0\|_2 = 1.$$

- No closed-form solution is found but analytical progress is possible close to criticality, via regular perturbation theory:

$$\mu_0 = \mu_{0c} + \epsilon^2 \Delta\mu, \quad 0 < \epsilon \ll 1, \quad \Delta\mu = \pm 1.$$

Introduction of nonlinear terms II

- In a regular perturbation theory in the small parameter ϵ , we find

$$\Omega = \omega_0 + \epsilon^2 \omega_1 + O(\epsilon^4),$$

where

$$\omega_1 = \frac{1}{\sqrt{\pi}} a |\chi| R^2 f(\theta, k^2), \quad k^2 = U^2 [\Re(1/\gamma)]^2 / 4 |\chi|^2, \quad \theta = \arg(\chi^2).$$

- Hence, a global mode (self-sustained nonlinear oscillation) exists for $R = R_c$, with

$$R_c^2 = -\epsilon^2 \frac{\Delta\mu}{\Im(\omega_1)}.$$

- Depending on the exact θ and k^2 , $\Im(\omega_1)$ will be positive and negative, and hence a global mode is ruled in / out for given θ and k^2 depending on the sign of $\Delta\mu$ – **parameter space**.

Stability of the global mode

By doing a secondary instability analysis, it can be shown analytically that the global mode (where it exists) is

- Unstable for a subcritical transition ($\Delta\mu = -1$)
- Stable for a supercritical transition ($\Delta\mu = +1$).

But all of this can be recast as a Stuart–Landay theory (?) – so what is new?

Higher-order perturbation theory

Going to higher order in the perturbation theory, e.g.

$$\Omega = \omega_0 + \omega_1 R^2 + \omega_2 R^4 + O(R^6),$$

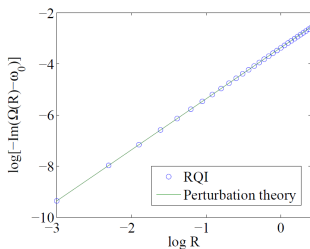
we see that at least two global modes are possible, with radii corresponding to the roots of

$$\Im \left[\omega_0 + \omega_1 R^2 + \omega_2 R^4 \right] = 0.$$

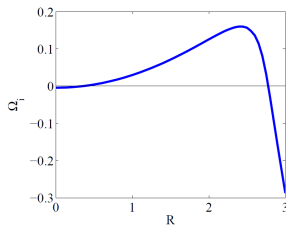
Even if the perturbation theory is not convergent this argument shows a possible mechanism for the existence of further global modes – and a motivation for numerical studies.

Two global modes

Sample RQI calculation indicates the presence of two global modes:



(e)



(f)

Numerical secondary instability analysis – the small- R global mode is unstable, while the large- R global mode **switches from unstable to stable as the advection U is increased.**

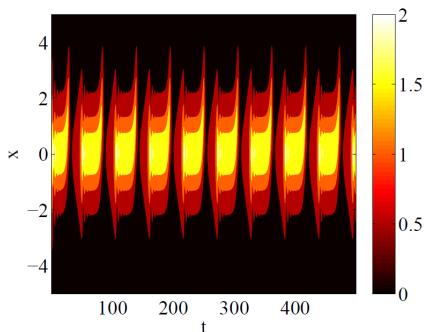
Would like to see how this behaviour is manifest in transient simulations.

Transient simulations – two unstable global modes I

Shown is spacetime plot of $|u(x, t)|$ for $U = 0$, corresponding to two unstable global modes. Initial condition:

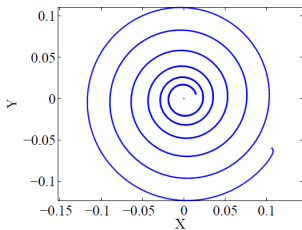
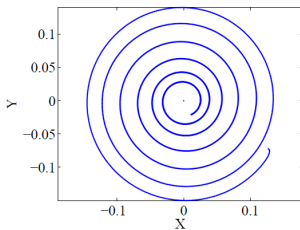
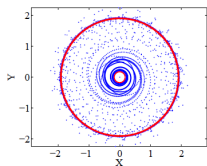
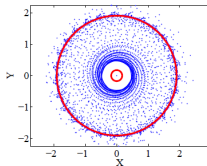
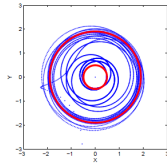
$$u(x, t = 0) = k_1 \psi_0(x)[1 + k_2 \psi_3(x)],$$

with $\rho = \|u(x, t = 0)\|_2$ a parameter ($\rho = 3$ below).

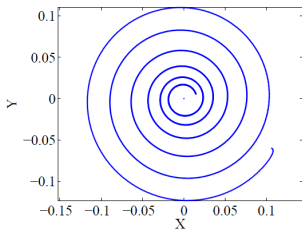


More useful to study $X(t) + iY(t) = u(0, t)$ as ρ varies.

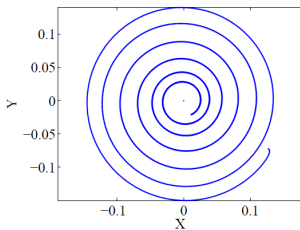
Transient simulations – two unstable global modes II

(a) $\rho = 0.2$ (b) $\rho = 0.3$ (c) $\rho = 0.4$ (d) $\rho = 3$ (e) $\rho = 100$

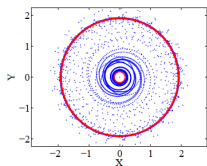
Transient simulations – two unstable global modes II



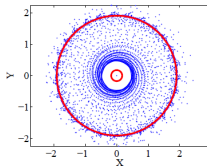
(a) $\rho = 0.2$



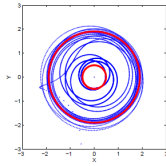
(b) $\rho = 0.3$



(c) $\rho = 0.4$



(d) $\rho = 3$



(e) $\rho = 100$

Transient simulations – two unstable global modes III

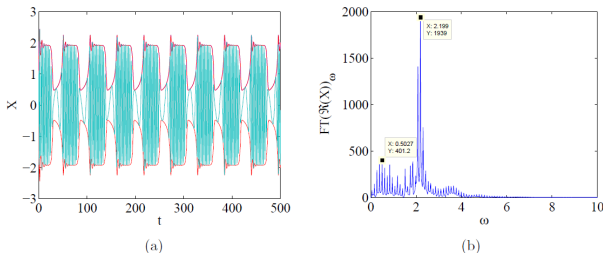


FIG. 10. Numerical simulations at $\rho = 3$. (a) Plot of $\Re(t)$ versus $X(t)$ showing the envelope curves; (b) Fourier transform of the same showing a single sharp maximum corresponding to the global-mode frequency at mode 2.

Numerical evidence is that the dynamics are **chaotic** with recurrent (but non-periodic) excursions away from the large-amplitude global mode.

Transient simulations – one unstable global mode

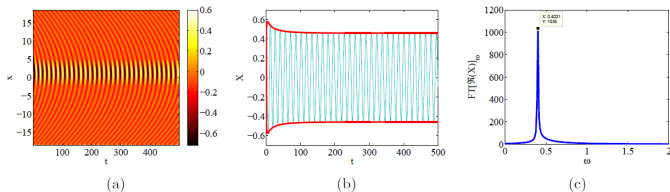


FIG. 12. DNS for the case $U = 1$. (a) Spacetime plot of $|u(x,t)|$; (b) Time series of $X = \Re(u(0,t))$; (c) Fourier transform of (b) showing the importance of the single large-amplitude global mode.

Numerical evidence is that the chaotic dynamics are suppressed as the large-amplitude liquid flows global mode is stabilized by advection.

Global modes – toy models: summary

- Simple dynamical models with nonlinearity and non-normality with exact and /or perturbative solutions

Global modes – toy models: summary

- Simple dynamical models with nonlinearity and non-normality with exact and /or perturbative solutions
- Leads to a better understanding of the interaction between transient growth (a linearized phenomenon) and the triggering of self-sustained nonlinear oscillations

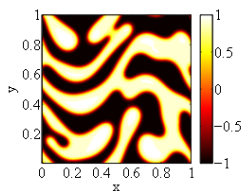
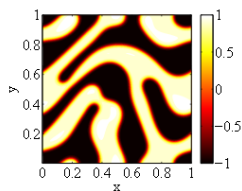
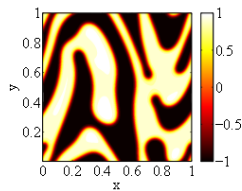
Global modes – toy models: summary

- Simple dynamical models with nonlinearity and non-normality with exact and /or perturbative solutions
- Leads to a better understanding of the interaction between transient growth (a linearized phenomenon) and the triggering of self-sustained nonlinear oscillations
- Of potential relevance in fluids to gain understanding of the subcritical transition

Global modes – toy models: summary

- Simple dynamical models with nonlinearity and non-normality with exact and /or perturbative solutions
- Leads to a better understanding of the interaction between transient growth (a linearized phenomenon) and the triggering of self-sustained nonlinear oscillations
- Of potential relevance in fluids to gain understanding of the subcritical transition
- Models are fully realised in optical context – third aim is *to investigate the possibility that fluid phenomena (transient growth, subcritical transition, self-sustained nonlinear behaviour) can be realised in an optical analogy, thereby opening up the possibility of having a **fluids lab on an optical bench**.*

*P*hase separation and the Cahn-Hilliard equation

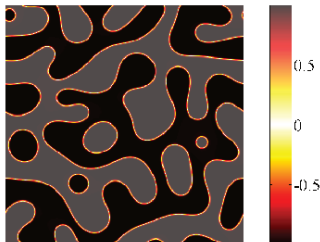
(a) $t = 5$ (b) $t = 10$ (c) $t = 15$

Phase separation

- Binary alloys phase separate below a critical temperature. The same phenomenon occurs for a wide range of mixtures.
- A mathematical model (first introduced for the binary alloys) for phase separation is the Cahn–Hilliard equation:

$$\frac{\partial c}{\partial t} = D \nabla^2 (c^3 - c - \gamma \nabla^2 c)$$

where c is the concentration field, D is the diffusion coefficient and $\sqrt{\gamma}$ is a length.



The solution is $c = \pm 1$ in domains with transition regions of thickness $\sqrt{\gamma}$ in between. The domains grow in time. The constant solution $c = 0$ is a well-mixed state but it is unstable.

The stirred Cahn–Hilliard equation

- The passive stirring a phase separated fluid is modelled by an advective term in the CH equation,

$$\frac{\partial c}{\partial t} + \mathbf{v} \cdot \nabla c = D \nabla^2 (c^3 - c - \gamma \nabla^2 c).$$

- The flow field can be prescribed ('passive stirring') or be obtained from coupling to the Navier–Stokes equations.
- Many studies in two dimensions, fewer in three. Our aim is to **fully characterize the structure of the domains under a generic three-dimensional flow.**

Direct numerical simulation of the Cahn–Hilliard equation

Numerical solution of the fourth-order Cahn–Hilliard equation with advection:

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = D \nabla^2 (c^3 - c - \gamma \nabla^2 c)$$

$$\nabla \cdot \mathbf{u} = 0,$$

Framework:

- Flow field \mathbf{u} is externally prescribed (e.g. synthetic turbulence)
- Periodic boundary conditions in each direction in **three dimensions**
- ‘Random’ initial conditions
 $c(\mathbf{x}, t = 0) = 0 + \text{random fluctuation}$

Numerical Discretization

- Very similar to TPLS – finite-volume method with a MAC grid for spatial discretization
- Combination of Adams–Bashforth and Crank–Nicholson methods for temporal discretization:

$$\frac{c_{ijk}^{n+1} - c_{ijk}^n}{\Delta t} = \left[\frac{23}{12} A_{ijk}^n - \frac{4}{3} A_{ijk}^{n-1} + \frac{5}{12} A_{ijk}^{n-2} \right] - \frac{1}{2} \gamma D \left(\nabla^4 c_{ijk}^{n+1} + \nabla^4 c_{ijk}^n \right),$$

$$A = \nabla^2 (c^3 - c) - \mathbf{u} \cdot \nabla c, \quad \alpha = \sqrt{\gamma D \Delta t}.$$

- Reduced to a linear problem at each timestep:

$$\left(1 + \gamma D \Delta t \nabla^4 \right) c_{ijk}^{n+1} = \text{RHS} := b_{ijk}.$$

Parallel Computing I

- Currently linear problem is solved by ‘operator factorization’ into a double Helmholtz problem:

$$\left(1 - \alpha \nabla^2\right)^2 c_{ijk}^{n+1} = b_{ijk} - 2\alpha \nabla^2 c_{ijk}^*, \quad c_{ijk}^* = c_{ijk}^{n+1}$$

and the unknown term on the right-hand side is estimated by interpolation:

$$c_{ijk}^* \approx \frac{3}{2} c_{ijk}^n - \frac{1}{2} c_{ijk}^{n-1}.$$

- The double Helmholtz problem can be solved by two passes of SOR:

$$\begin{aligned} \left(1 - \alpha \nabla^2\right) c_{ijk}^{n+1} &= \omega_{ijk}, \\ \left(1 - \alpha \nabla^2\right) \omega_{ijk} &= b_{ijk} - 2\alpha \nabla^2 c_{ijk}^* \end{aligned}$$

- As in standard diffusive problems, the matrix is diagonally-dominant and convergence is rapid.

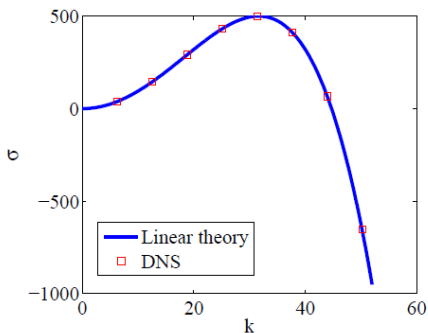
Parallel Computing II

- Code is parallelized using full 3D MPI domain decomposition
- Periodic boundary conditions in each dimension – handled effortlessly by MPI
- SOR is modified to include ‘red-black colouring’ in the parallel version – eliminates ‘race conditions’
- For typical problem size – 312^3 gridpoints – domain decompositions fit into the cache on Fionn, meaning that superlinear speedup has been achieved in the code with up to ~ 500 MPI processes.

Strict benchmarks for code's accuracy (again)

- The equilibrium $c = 0$ is linearly unstable to small-amplitude perturbations of the form $\delta c \propto e^{\sigma t + i\mathbf{k} \cdot \mathbf{x}}$.
- In the absence of flow the dispersion relation is known:

$$\sigma(k) = Dk^2(1 - \gamma k^2), \quad k = |\mathbf{k}|.$$



Comparison between linear theory and direct numerical simulation. Model parameters: $D = 1$ and $\gamma = 5 \times 10^{-4}$. Simulation parameters: $\Delta x = 1/304$, $\Delta t = 10^{-6}$.

Very simple turbulence model

Prescribed flow to mimic turbulence:

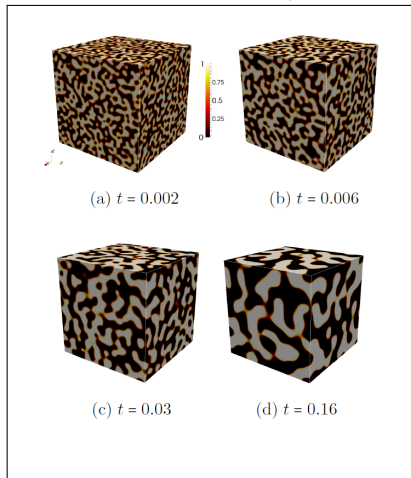
$$\begin{aligned}
 u &= A \sin(ky + kz + \varphi_i), & 0 \leq \text{mod}(t, \tau) < \frac{1}{3}\tau, \\
 v &= A \sin(kx + kz + \psi_i), & \frac{1}{3}\tau \leq \text{mod}(t, \tau) < \frac{2}{3}\tau, \\
 w &= A \sin(kx + ky + \chi_i), & \frac{2}{3}\tau \leq \text{mod}(t, \tau) < \tau,
 \end{aligned}$$

where the other velocity components are zero, and

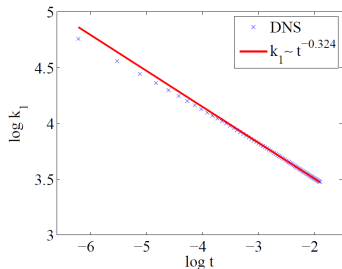
- τ is the quasi-period of the flow and $t = q\tau + \mu$ with q an integer and $0 \leq \mu < \tau$, hence $\mu = \text{mod}(t, \tau)$.
- The random phases (φ, ψ, χ) are renovated once every $N\tau$ periods – correlation time $\tau_{\text{corr}} = N\tau$.
- Flow is manifestly incompressible, $\nabla \cdot \mathbf{u} = 0$.
- Analogous model for 2D can be written down.

Some results – phase separation without flow

In the absence of flow, domains grow algebraically in time.

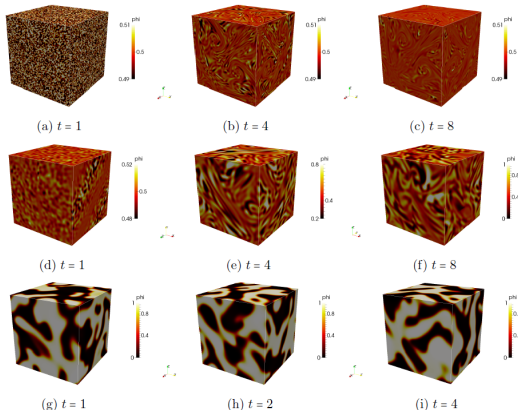


Domains coarsen at a rate $\ell \sim t^{1/3}$ – Lifshitz–Slyozov law in 3D.



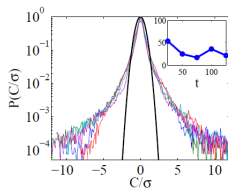
Some results – phase separation with flow

- Shear flow induces coarsening arrest.
- At large flow amplitudes and **long correlation times**, domains are overwhelmed and *remixing* occurs.
- This limit can be described by linear theories (e.g. advection-diffusion).

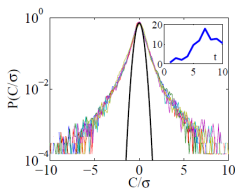


Importance of dimensionality

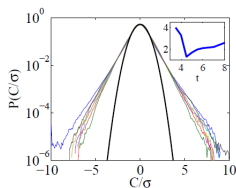
Dimensionality manifests itself in the remixing limit through the PDF of the concentration – first two figures in 2D, last figure on RHS in 3D.



(a)



(b)



(c)

Reason:

- Apply effective-diffusion theory to linearized equations.
- Green's function has longer range in 2D compared to 3D
- Leads to a distribution with 'fat' tails

Conclusions

- TPLS and its derivatives can be used to simulate a variety of two-phase flow applications.
- Best to have some theory to make sense of the enormous amounts of data produced by the simulations!
- TPLS is open source!

The screenshot shows a web browser window displaying the SourceForge project page for TPLS. The browser's address bar shows the URL `sourceforge.net/projects/tpls/`. The page header includes navigation links: Home / Browse / Science & Engineering / Scientific Engineering / TPLS. The main heading is "TPLS" with the subtitle "High Resolution Direct Numerical Simulation (DNS) of Two-Phase Flows" and the text "Brought to you by: ibethune, onaraigh, pvaluri". Below the heading are tabs for Summary, Files, Reviews, Support, Tickets, Code, and Wiki. A "Download" button is visible, along with social media sharing options for Twitter, Google+, and Facebook. Three 3D simulation images are displayed in a row: a long channel with a blue liquid phase, a complex 3D structure with blue liquid, and a red liquid phase. The right sidebar contains a list of related products: ANSYS Simulation Software, 100% Online MBA Degree, Spell Check Download, Social Studies Course, and Free Driver Updates.