Coarsening dynamics of binary liquids with active rotation

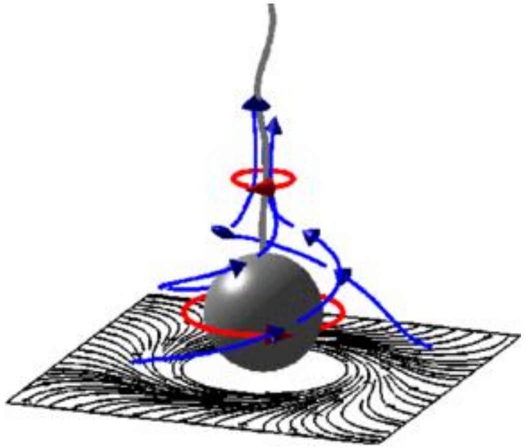
PHYS 230

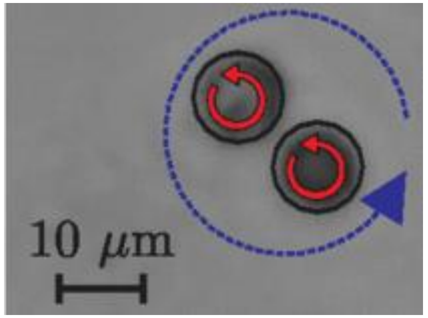
Presented by Patrick Noerr and Joshua Tamayo

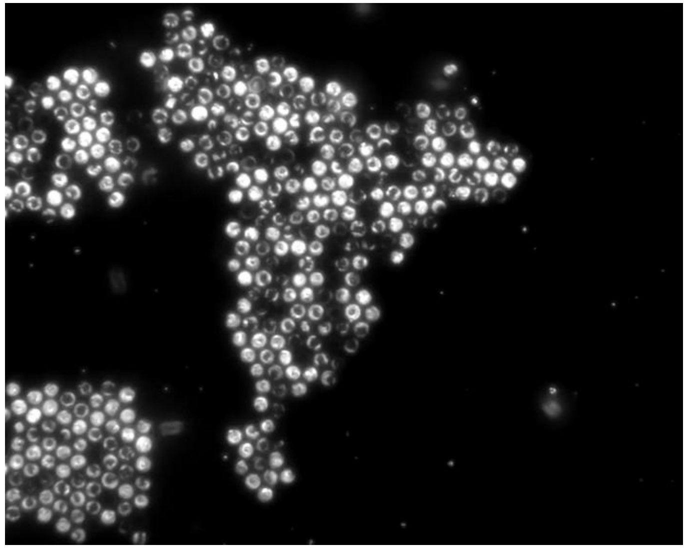
May 6, 2021

Sabrina, S., Spellings, M., Glotzer, S. C., & Bishop, K. J. M. (2015). Coarsening dynamics of binary liquids with active rotation. *Soft Matter*, *11*(43), 8409–8416. doi:10.1039/c5sm01753j

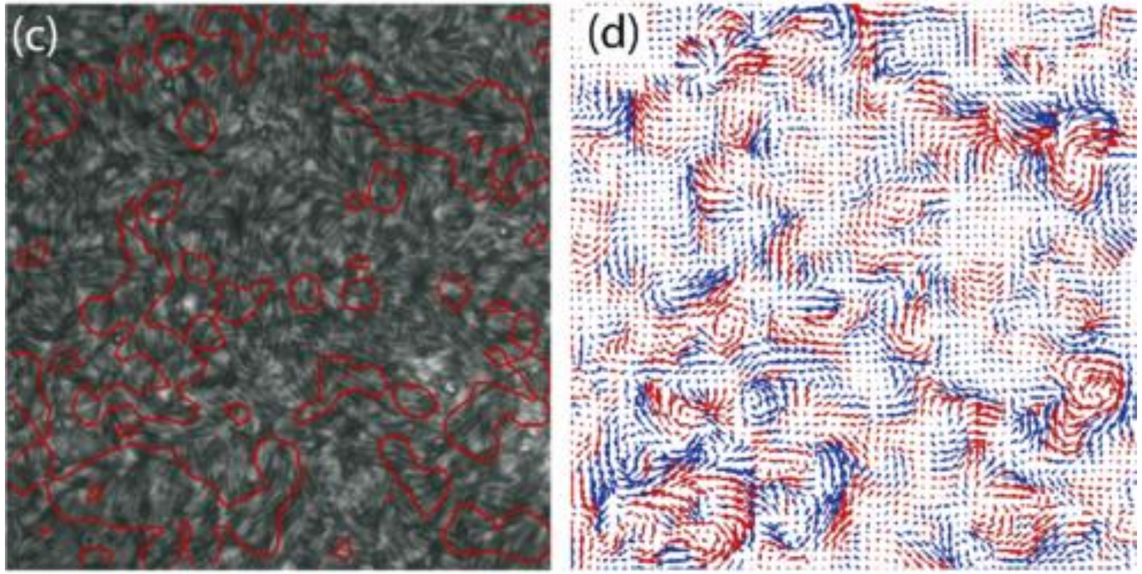
Active matter and collective behavior







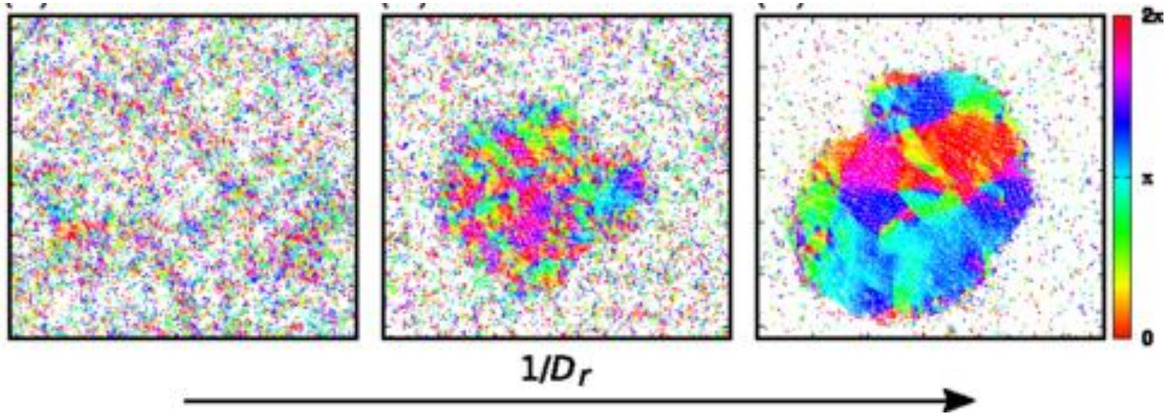
Clusters of actively rotating *Thiovulum majus* generate vortex-like flows that attract other cells [**1**]



Sublethal antibiotic concentrations generate motile and immotile regions of *B. subtilis*. Red circles denote immotile clusters (Motility-Induced Phase Separation) [**2**]

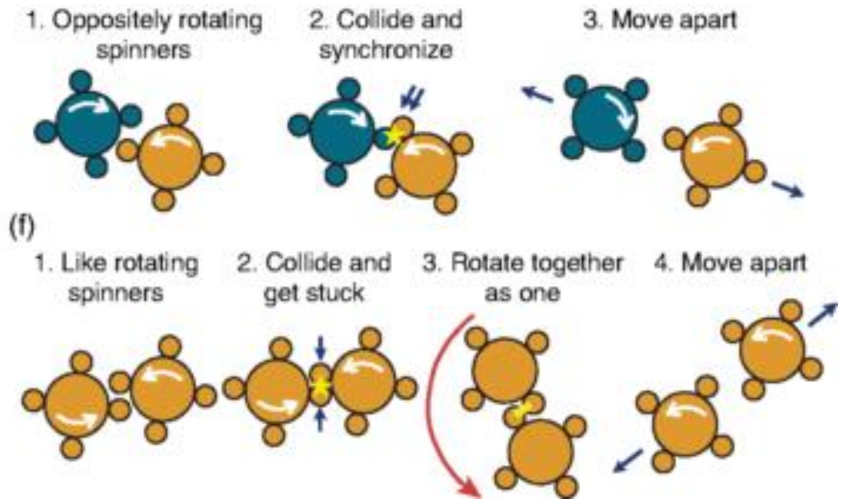
[1] Petroff AP, Wu XL, Libchaber A. Fast-moving bacteria self-organize into active two-dimensional crystals of rotating cells. Phys Rev Lett. 2015 Apr 17;114(15):158102. doi: 10.1103/PhysRevLett.114.158102. Epub 2015 Apr 17. PMID: 25933342.
[2] Benisty, S., Ben-Jacob, E., Ariel, G., & Be’er, A. (2015). Antibiotic-Induced Anomalous Statistics of Collective Bacterial Swarming. Physical Review Letters, 114(1), 018105. doi:10.1103/physrevlett.114.018105

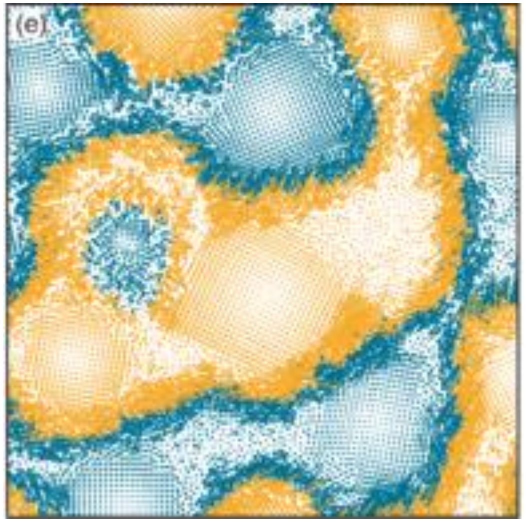
Active particles phase separate in agent-based models



Active Brownian particles phase separate at sufficient densities and low influence from rotational diffusion [**3**]

Previous work observed simulations of counter-rotating, hard spinners. <u>Steady convective flows originate along phase interface that influence coarsening</u>. [**4**]

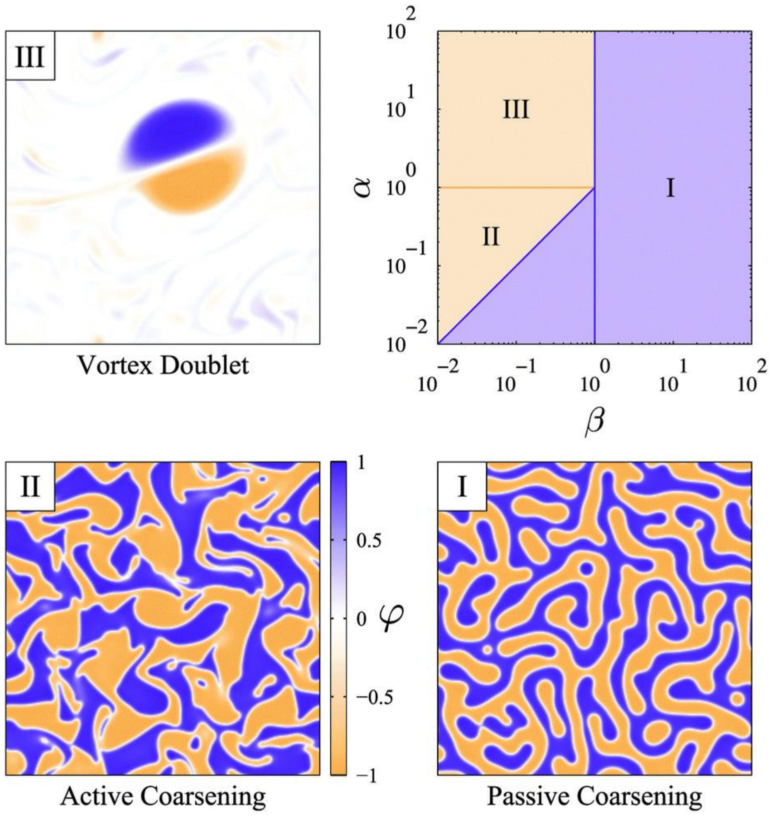




[3] Caprini, L., Marconi, U. M. B., & Puglisi, A. (2020). Spontaneous Velocity Alignment in Motility-Induced Phase Separation. *Physical Review Letters*, *124*(7), 078001. doi:10.1103/physrevlett.124.078001

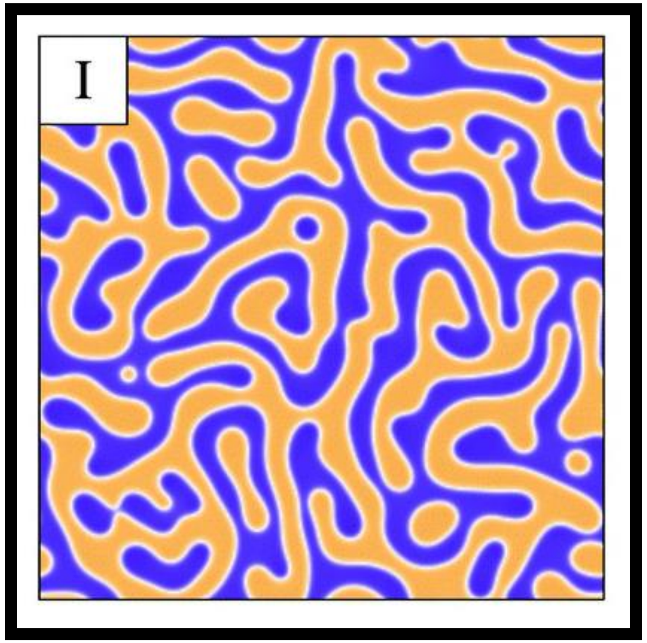
[4] Nguyen, N. H. P., Klotsa, D., Engel, M., & Glotzer, S. C. (2013). Emergent Collective Phenomena in a Mixture of Hard Shapes through Active Rotation. *Physical Review Letters*, *112*(7), 075701. doi:10.1103/physrevlett.112.075701

Key objectives

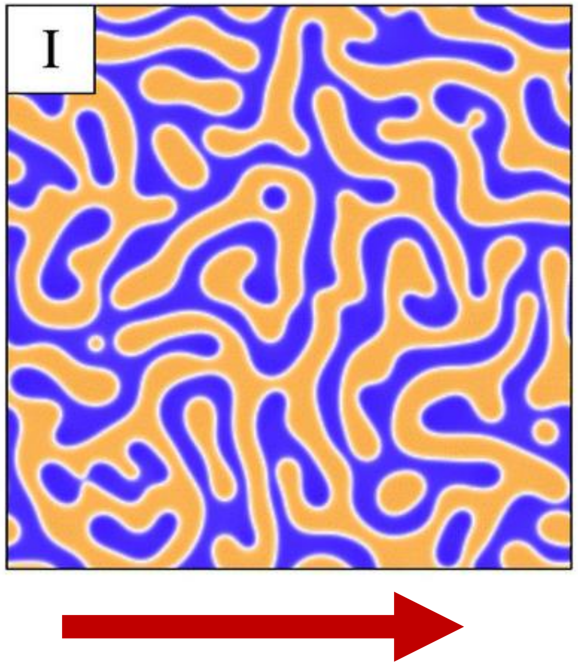


Check: See if our model can replicate the phase-diagram

Use a continuum model of counter-rotating liquid phases to study liquid coarsening



Bounded domains or semi-periodicity



Flow in a certain direction

Governing equations

The convective Cahn-Hilliard equation describes the process of phase separation with the addition of convective flow terms

M $\rightarrow$ Mobility
$\mathbf{v} \rightarrow$ Fluid velocity
$\lambda, K, r \rightarrow$ Positive coefficients determining thickness of phase interface
$\phi \rightarrow$ Order parameter defining phases

Convective Cahn-Hilliard equation

$$\frac{\delta \phi}{\delta t} + \nabla \cdot (\phi \mathbf{v}) = \mathrm{M}\nabla^2(\lambda\phi^3 - K\nabla^2\phi - r\phi)$$

The Navier-Stokes equation describes the activity-driven fluid flow in the continuum model with the addition of capillary forces, active rotation, and frictional drag

$\rho \rightarrow$ Fluid density
p $\rightarrow$ Pressure
$\eta \rightarrow$ Fluid viscosity
$\phi \rightarrow$ Order parameter defining phases

2D Incompressible Navier-Stokes Equation

$$\rho \frac{\mathrm{d}\mathbf{v}}{\mathrm{dt}} = -\nabla p + \eta\nabla^2\mathbf{v} + \mu\nabla\phi + \nabla \times (\phi\boldsymbol{\tau}) - b\mathbf{v}$$

$$\nabla \cdot \mathbf{v} = 0$$

Governing equations

Characteristic scales

Interface thickness $\rightarrow \left(\frac{\mathrm{K}}{\mathrm{r}}\right)^{\frac{1}{2}}$

Unmixing time $\rightarrow \frac{\mathrm{K}}{\mathrm{Mr}^2}$

Equilibrum composition $\rightarrow \left(\frac{\mathrm{r}}{\lambda}\right)^{\frac{1}{2}}$

Chemical potential $\rightarrow \left(\frac{\mathrm{r}^3}{\lambda}\right)^{\frac{1}{2}}$

Key Parameters

$\boldsymbol{\alpha} \rightarrow$ Strength of active rotation

$\boldsymbol{\beta} \rightarrow$ Strength of frictional drag

Non-dimensional convective Cahn-Hilliard equation

$$\frac{\delta\phi}{\delta t} + \mathbf{v}\cdot\nabla\phi = \nabla^2(-\phi + \phi^3 - \nabla^2\phi)$$

Non-dimensional Navier-Stokes

$$\mathrm{Re}\frac{\mathrm{d}\mathbf{v}}{\mathrm{dt}} = -\nabla p + \nabla^2\mathbf{v} + \mathrm{Ca}^{-1}\mu\nabla\phi + \alpha\nabla\times(\phi\boldsymbol{e}_z) - \beta\mathbf{v}$$

Low Reynolds number (Re): $\mathrm{Re} = \frac{\rho M r}{\eta} \rightarrow 0$

Neglect capillary forces: $\mathrm{Ca}^{-1} \rightarrow 0$

$$0 = -\nabla p + \nabla^2\mathbf{v} + \alpha\nabla\times(\phi\boldsymbol{e}_z) - \beta\mathbf{v}$$

Semi-implicit Fourier spectral method

- Governing equations solved in Fourier space [**4**]
  - We solve the velocity field in Fourier space, and the composition (φ) in real space
- Explicit schemes have severe time-step constraints
- 2nd order schemes converge faster

Cahn-Hilliard Equation

$$\frac{\delta\Phi(k,t)}{\delta t} + \left(ik_x\{v_x\phi\}_k + ik_y\{v_y\phi\}_k\right) = -k^2\{\phi^3 - \phi\}_k - k^4\Phi(k,t)$$

Navier-Stokes Equation

$$0 = k^4\Psi(k,t) - \boldsymbol{\alpha}k^2\Phi(k,t) + \boldsymbol{\beta}k^2\Psi(k,t)$$

To numerically solve these equations, the stream function (ψ) is included where $v = \nabla \times (\psi\boldsymbol{e}_z)$

[4] J. Zhu, L. Q. Chen, J. Shen and V. Tikare,Phys. Rev. E, 1999,60, 3564–3572.

Semi-implicit Fourier spectral method

- Governing equations solved in Fourier space [**4**]
  - We solve the velocity field in Fourier space, and the composition (φ) in real space
- Explicit schemes have severe time-step constraints
- 2nd order schemes converge faster

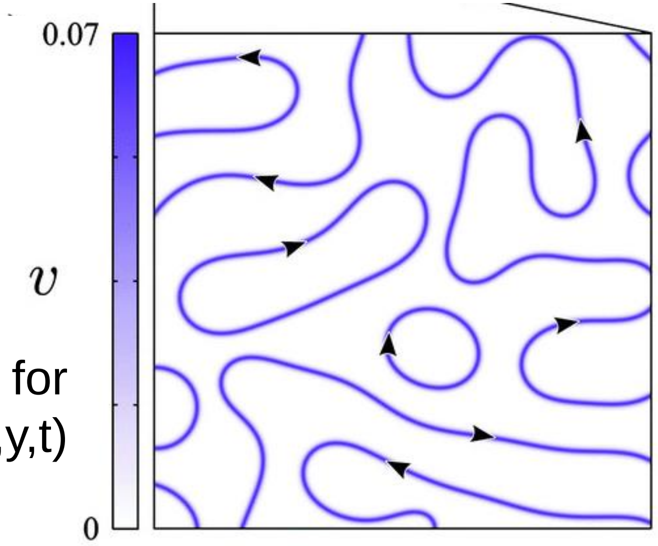
Cahn-Hilliard Equation

$$\frac{\delta\Phi(k,t)}{\delta t} + \left(ik_x\{v_x\phi\}_k + ik_y\{v_y\phi\}_k\right) = -k^2\{\phi^3 - \phi\}_k - k^4\Phi(k,t)$$

Stream-function

$$0 = \nabla^4\psi + \boldsymbol{\alpha}\nabla^2\phi - \boldsymbol{\beta}\nabla^2\psi$$

To numerically solve these equations, the stream function (ψ) is included where $v = \nabla \times (\psi \boldsymbol{e}_z)$



Plot of the stream function for a given v(x,y,t)

[4] J. Zhu, L. Q. Chen, J. Shen and V. Tikare,Phys. Rev. E, 1999,60, 3564–3572.

Time step simplification

- We solve the velocity field in Fourier space, and the composition (φ) in real space
- We simplify the time stepping by using forward-Euler method

<u>Scheme</u>

$$\frac{\delta\phi}{\delta t} + \mathbf{v}\cdot\nabla\phi = \nabla^2(-\phi + \phi^3 - \nabla^2\phi)$$

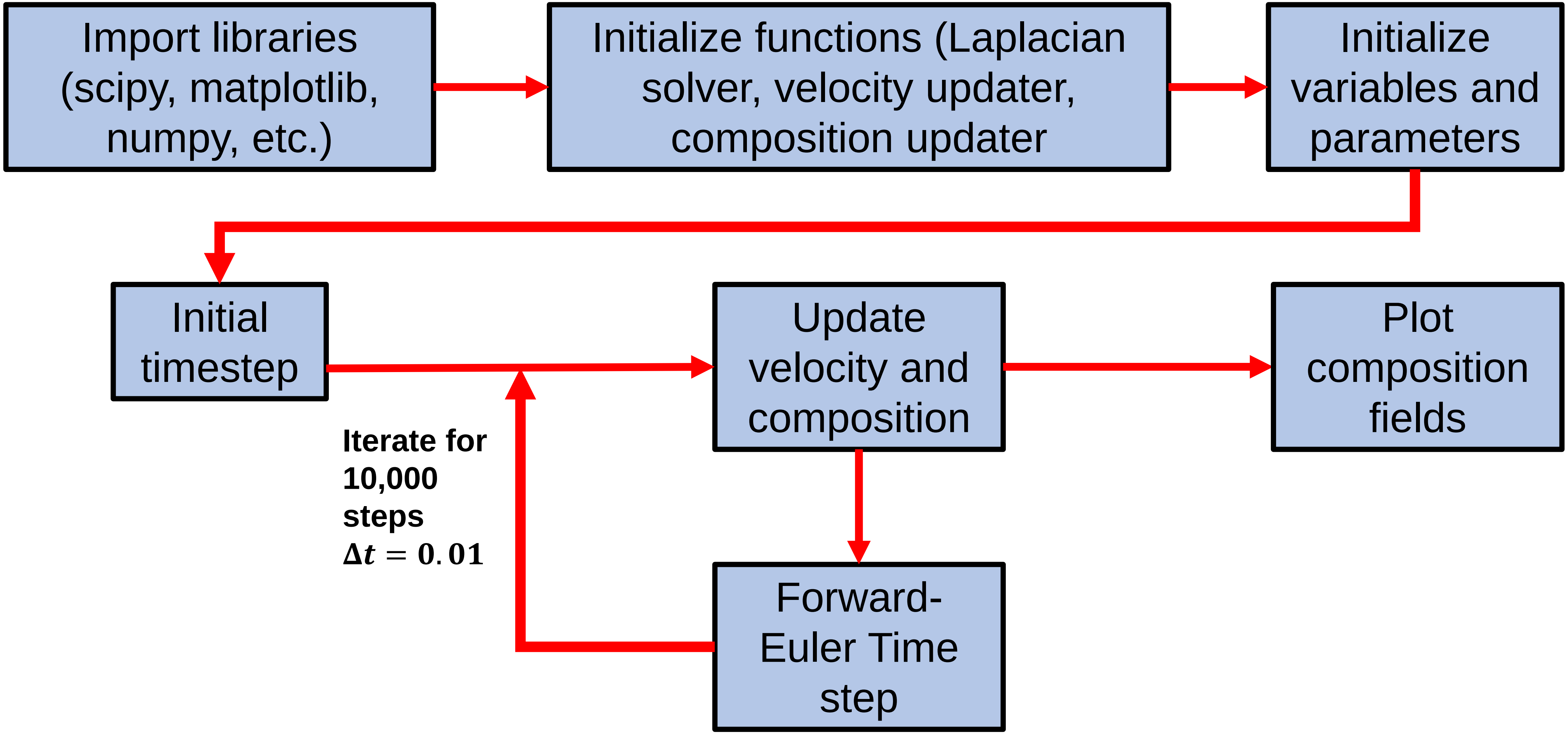
Update compositional order parameter (ϕ)

$$\Psi(\mathbf{k},t) = \frac{\alpha}{k^2+\beta}\boldsymbol{\Phi}(\mathbf{k},\mathrm{t})$$

$$V_x = ik_y\Psi \qquad V_y = -ik_x\Psi$$

Inverse FFT to obtain velocity in real space

Computational workflow



Computational workflow continued

Import libraries (scipy, matplotlib, numpy, etc.)

```
%matplotlib inline
import numpy as np
import random
import os
from scipy.fft import fft2, ifft2, fftfreq
```

Parameter/Variable	Value
Grid points (N)	100
x-dir grid spacing (dx)	~ 0.02
y-dir grid spacing (dy)	~ 0.02
# of time steps (nt)	10^4
Time step (dt)	10^{-2}

Computational workflow continued

Initialize functions (Laplacian solver, velocity updater, composition updater

```python
def discrete_laplacian(M):
    """Get the discrete Laplacian of matrix M"""
    L = -4*M
    L += np.roll(M, (0,-1), (0,1)) # right neighbor
    L += np.roll(M, (0,+1), (0,1)) # left neighbor
    L += np.roll(M, (-1,0), (0,1)) # top neighbor
    L += np.roll(M, (+1,0), (0,1)) # bottom neighbor
    return L
```

```python
def phi_update(u,VX,VY,t):
    u = np.asarray(u)
    clone = u.copy()
    u = clone + (-(np.multiply(VX,(np.gradient(clone))[0])
                   +np.multiply(VY,(np.gradient(clone))[1]))
                 +discrete_laplacian(-clone + clone**3. - discrete_laplacian(clone)))*t

    return u
```

Computational workflow continued

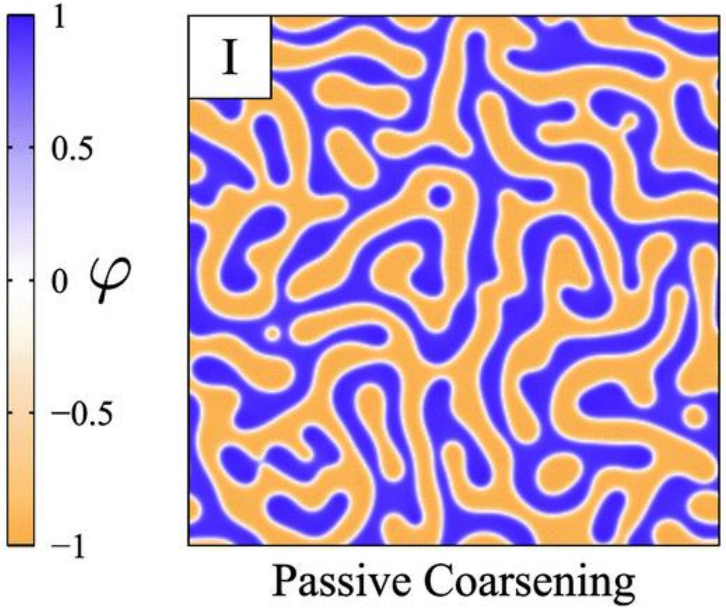
Initialize functions (Laplacian solver, velocity updater, composition updater

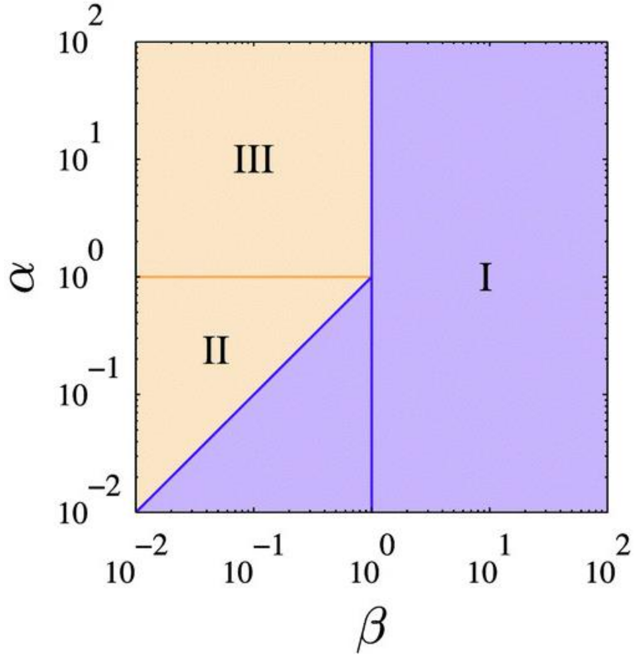
```python
def get_initial_configuration(N):
    dx = 2/N
    dy = 2/N
    u=[[random.uniform(-.1,.1) for i in range(N)] for j in range(N)]
    return u
```

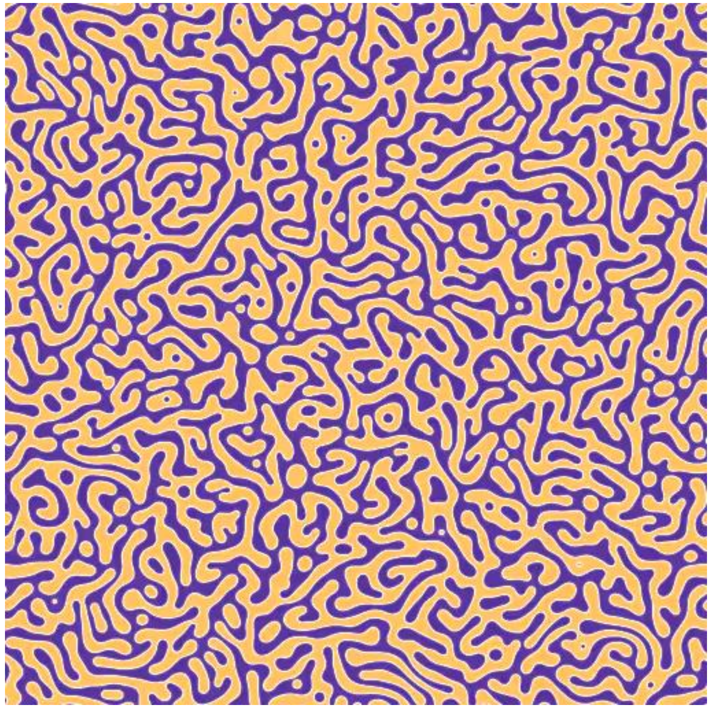
```python
def velocity(u,a,b,dx,dy,N):
    psi = np.ones((N,N))
    velx = np.ones((N,N))
    vely = np.ones((N,N))
    uprime = fft2(u)
    n_value = fftfreq(N,(1.0/N))
    kx_array = np.zeros((N,N),dtype = float)
    ky_array = np.zeros((N,N),dtype = float)
    x_length = 2
    y_length = 2
    for i in range(0,N):
        for j in range(0,N):
            kx_array[i][j]=(2*np.pi*n_value[j])/x_length
            ky_array[i][j]=(2*np.pi*n_value[i])/y_length
            velx[i][j] = uprime[i][j]*complex(1)*kx_array[i][j]*(a/((kx_array[i][j]**2.+ky_array[i][j]**2.)+b))
            vely[i][j] = uprime[i][j]*(-complex(1))*ky_array[i][j]*(a/((kx_array[i][j]**2.+ky_array[i][j]**2.)+b))
    velx, vely = np.real(ifft2(velx)), np.real(ifft2(vely))
    return velx, vely
```

```python
#This is taken from PHYS 230 notebook "PhaseSeparationModel.ipynb"
def forward_solver_full(a,b):
    N = 100
    dx = 2.0 / N
    dy = 2.0 / N
    x = np.linspace(-1.0, 1.0, N)
    y = np.linspace(-1.0, 1.0, N)
    Y, X = np.meshgrid(y, x)
    u = get_initial_configuration(N)
    fig = plt.figure()
    axes = fig.add_subplot(1, 1, 1, aspect='equal')
    plot = plt.imshow(u, extent=(-1, 1, -1, 1), interpolation='nearest')
    fig.colorbar(plot)
    t = 0.0
    dt = .01
    num_steps = 10000
    next_output_time = 0.0
    for j in range(num_steps):
        velx, vely = velocity(u, a, b, dx, dy, N)
        u = phi_update(u,velx,vely,dt)
        t += dt
        if t >= next_output_time:
            next_output_time += 1
            fig1 = plt.figure()
            axes1 = fig1.add_subplot(1, 1, 1, aspect='equal')
            plot1 = plt.imshow(u, extent=(-1, 1, -1, 1), interpolation='bilinear')
            fig1.colorbar(plot1)
            plt.show()
    plt.show()
forward_solver_full(a=10, b=100)
```

Passive coarsening







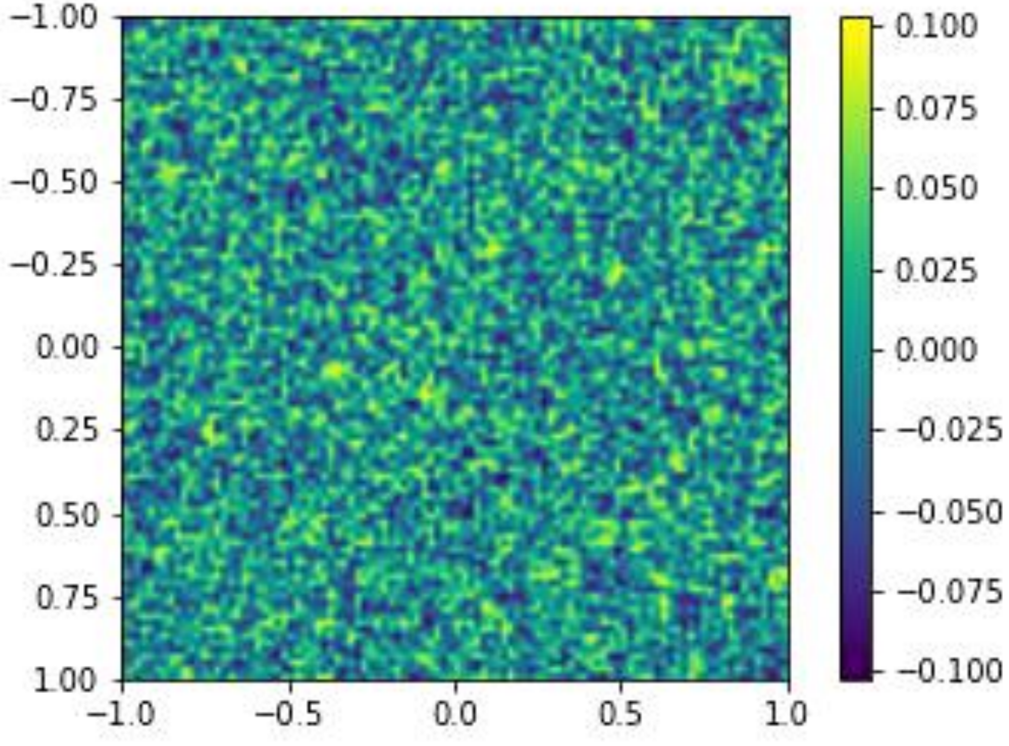
Passive coarsening is observed for strong frictional drag (β >> 1) such that convection is negligible

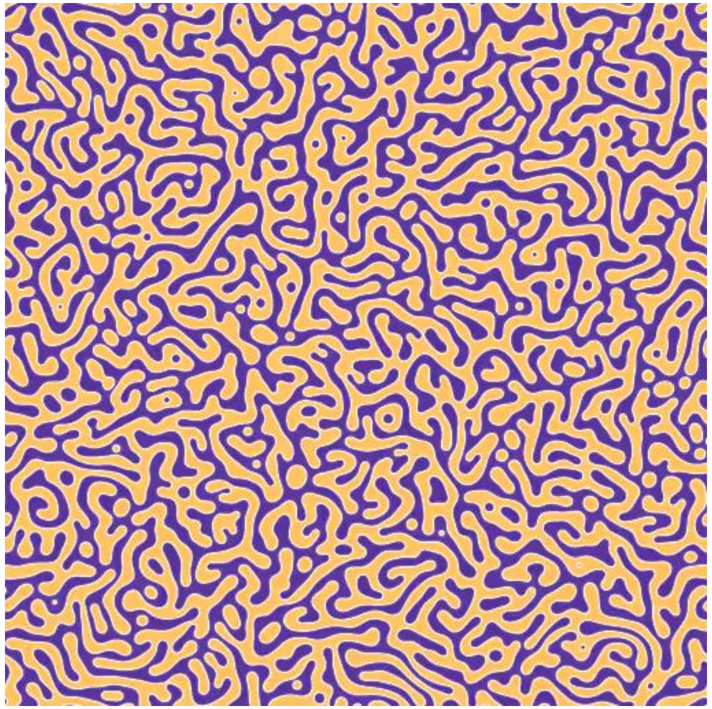
This system has been previous studied using only the convective Cahn-Hilliard equation

Fluid flow becomes negligible as the force of active rotation are balanced by friction

Passive coarsening

Passive Coarsening ($\alpha = 10, \beta = 100$)

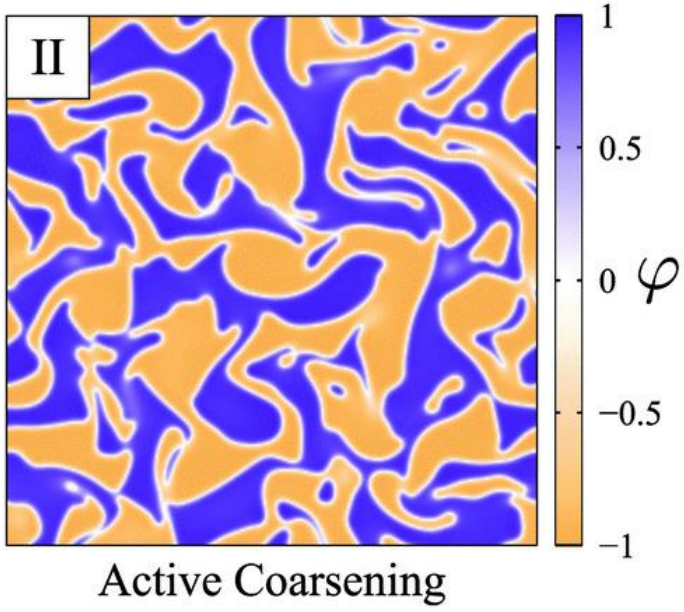


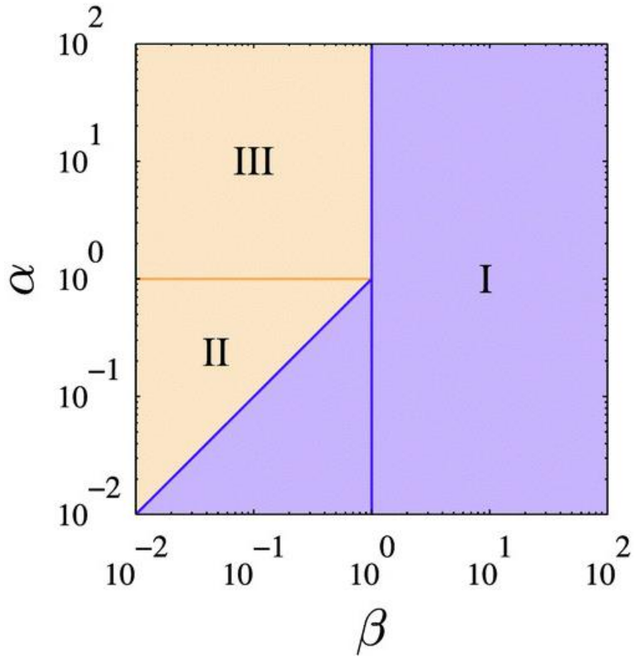


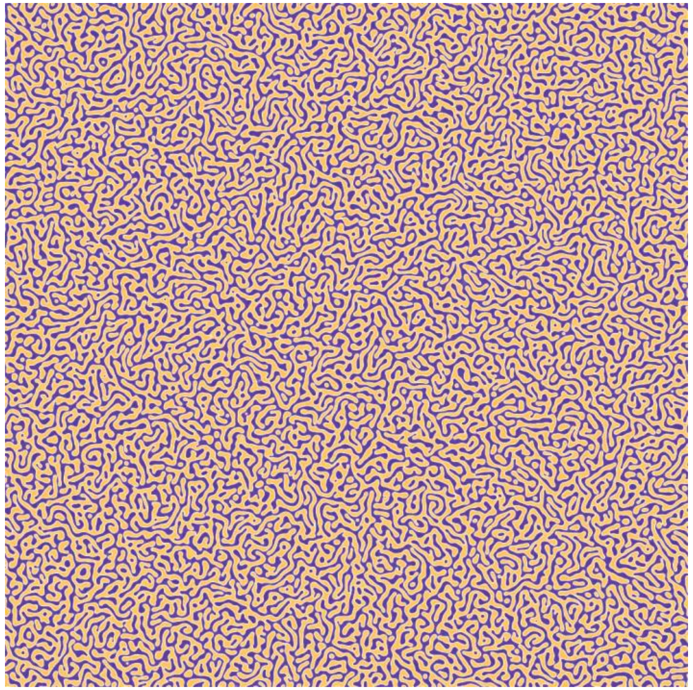
Passive coarsening is observed for strong frictional drag (β >> 1) such that convection is negligible

Fluid flow becomes negligible as the force of active rotation are balanced by friction

Active coarsening





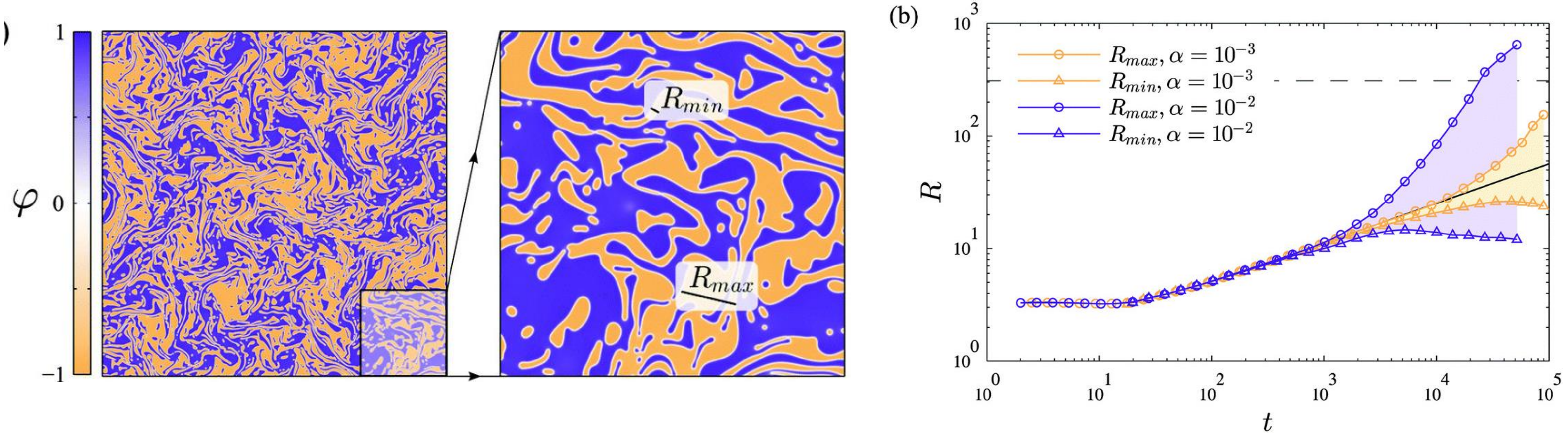


Active coarsening is observed for weak frictional drag and weak rotation (β << 1, α << 1)

Flows due to active rotation extend into the bulk as opposed to remaining along the interface. Thin filaments form that are absorbed into larger domains.

Domain sizes grow faster than with just passive coarsening

Active coarsening (no frictional dampening)

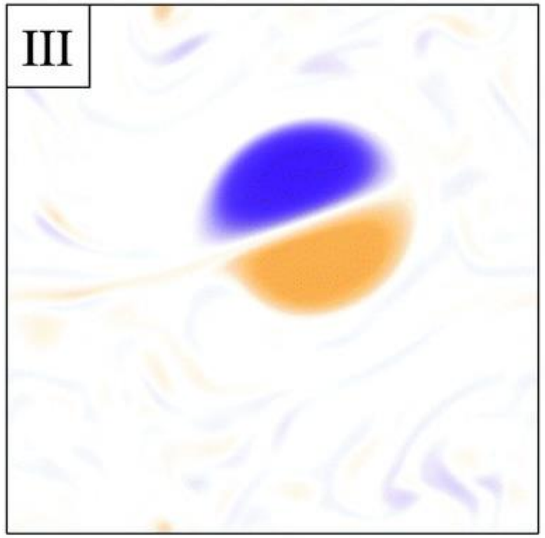


Flows due to active rotation extend into the bulk as opposed to remaining along the interface. Thin filaments form that are absorbed into larger domains.

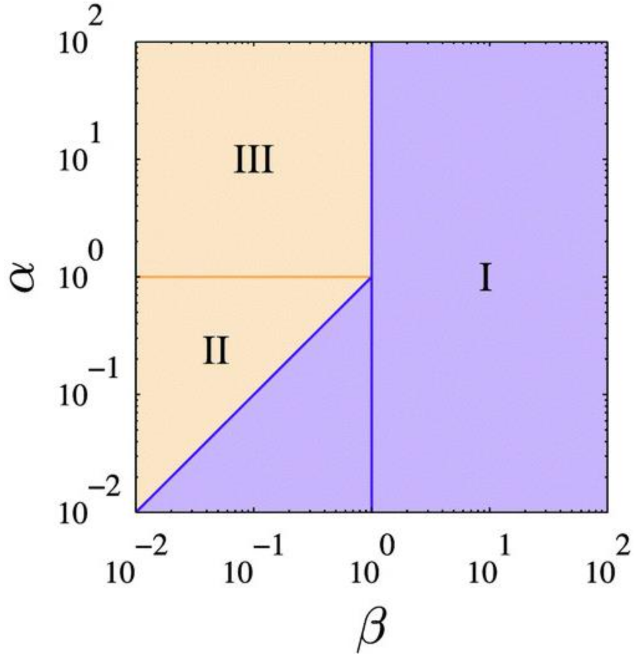
They observe a range of domain sizes as opposed the same domain sizes throughout the liquid, but continuous mixing occurs due to absence of frictional drag

Active coarsening is observed for weak frictional drag and weak rotation (β = 0, α << 1)

Vortex Doublet



Vortex Doublet

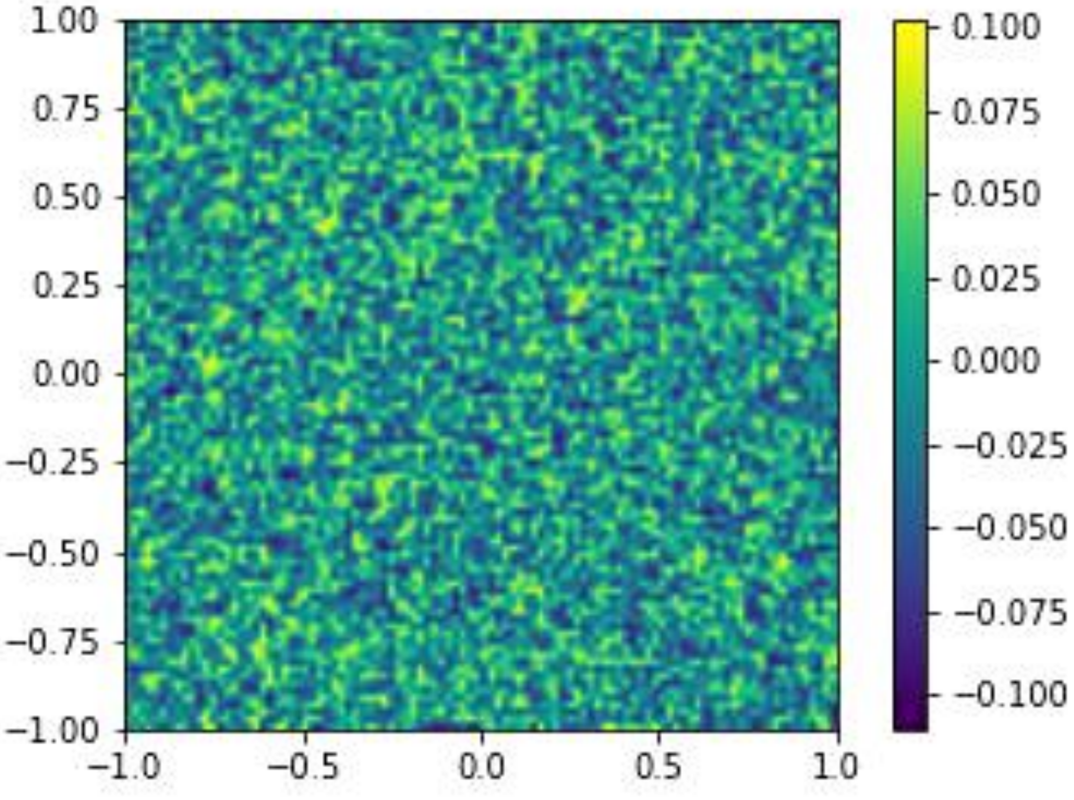


Vortex doublets are observed for no frictional drag and strong rotation (β = 0, α >> 1)

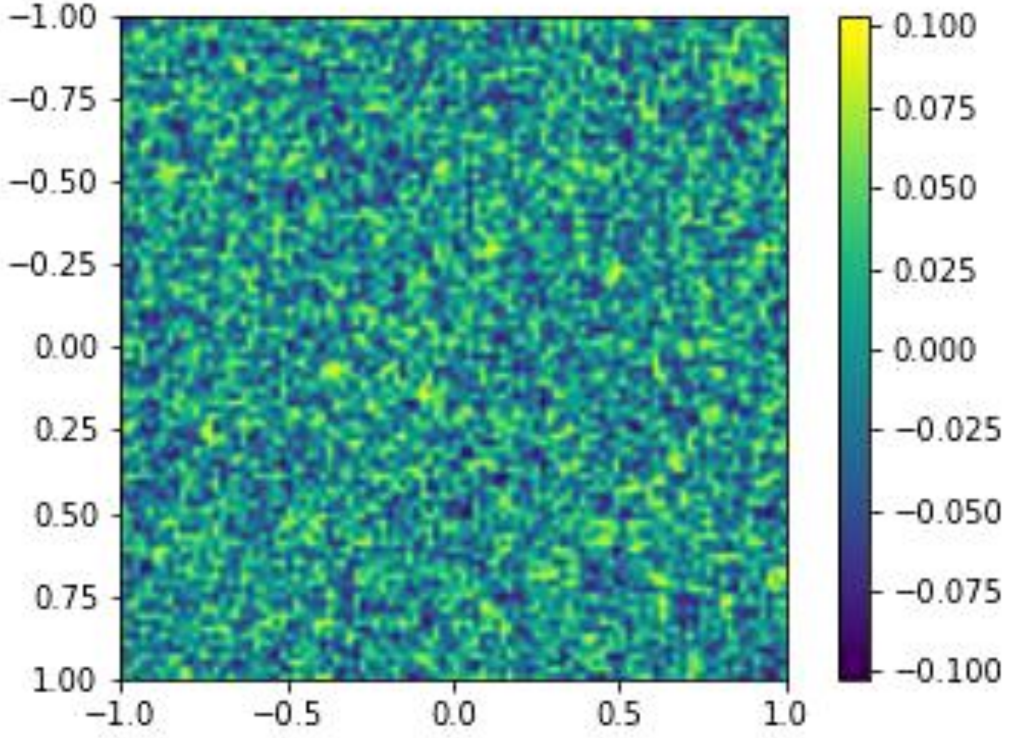
Local phases begin to form early. These small phases are of high vorticity and low shearing that promote phase separation.

Some vortices are broken due to the shear, others merge into large vortices

Aperiodic results

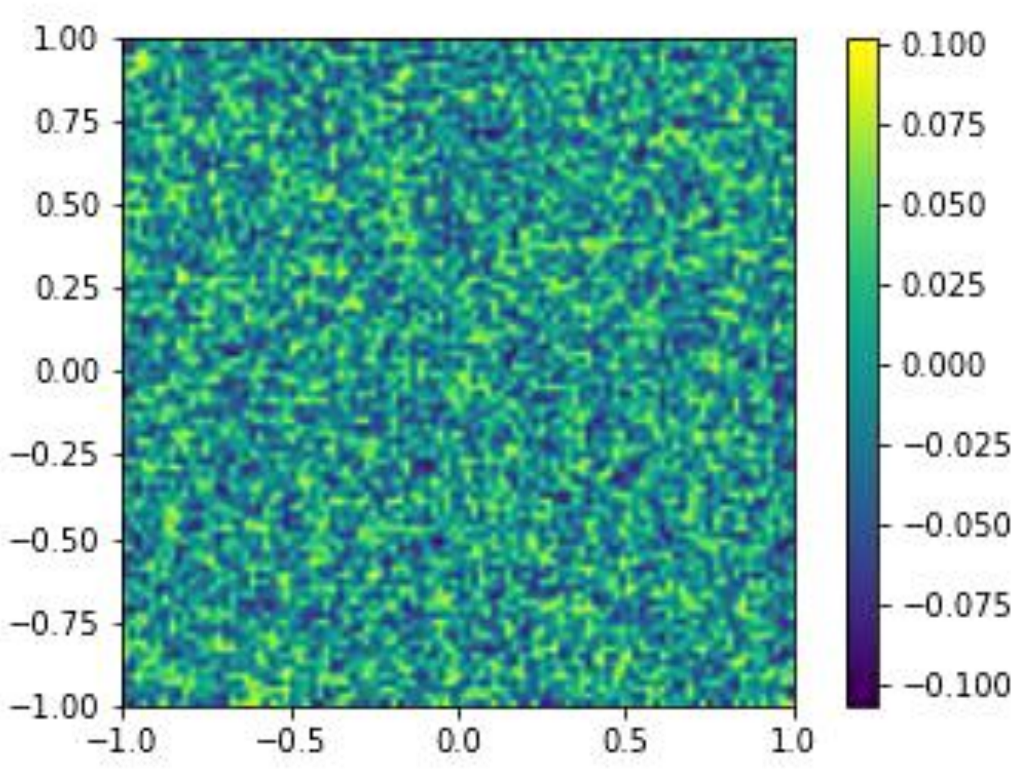


Aperiodic boundaries

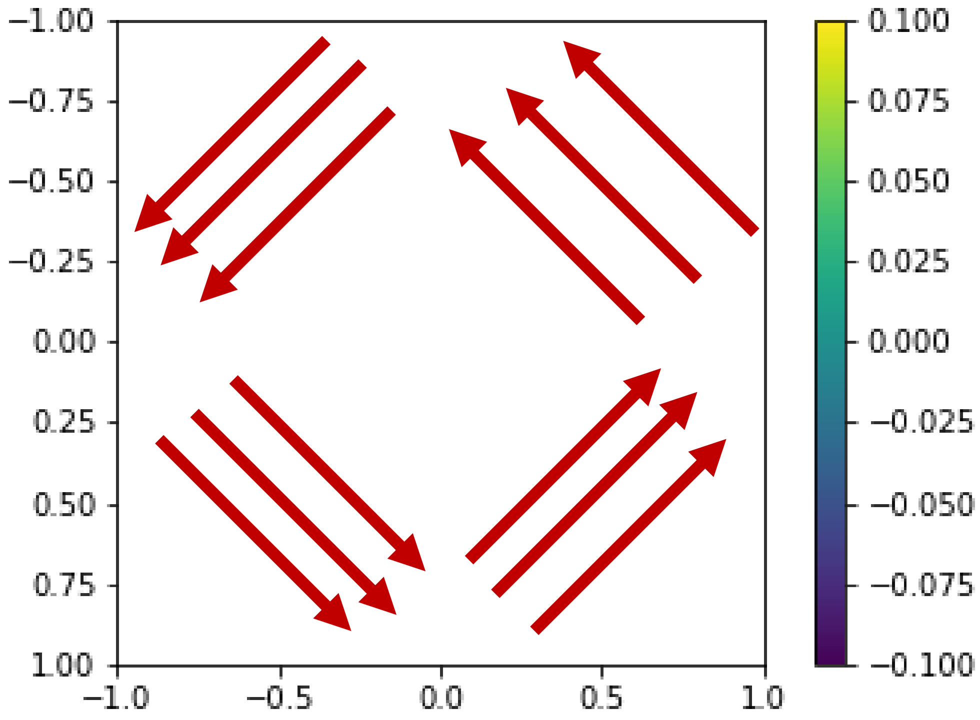


Periodic boundaries

Swirl Flow



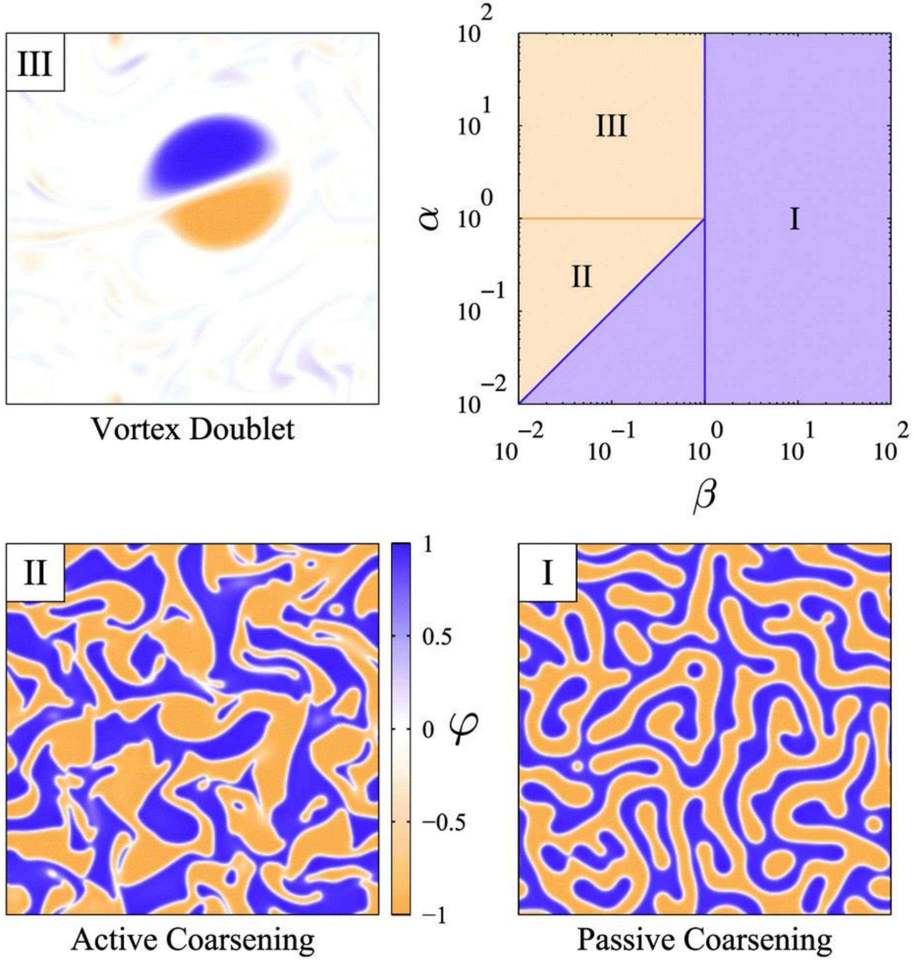
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.0
−0.5
0.0
0.5
1.0
0.100
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.075
−0.100



−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
−1.0
−0.5
0.0
0.5
1.0
0.100
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.075
−0.100

Future Plans

- Fix velocity fields
  - Implement the semi-periodic Fourier spectral method
- Add directional flow and see how that influences the domain sizes and mixing



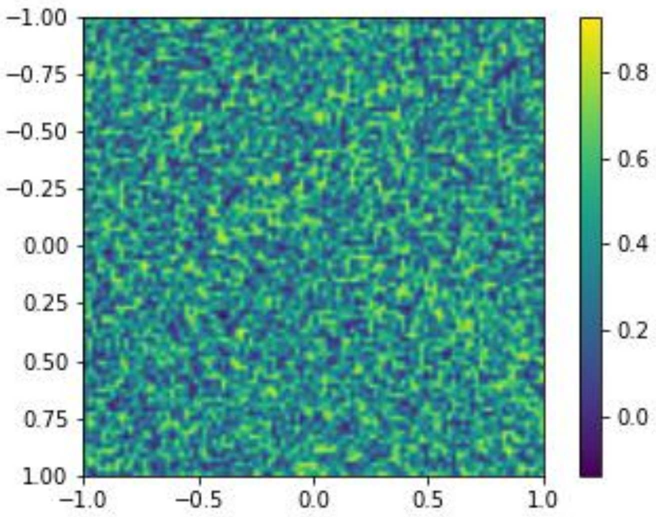
Additional results

As of 5/21/2021

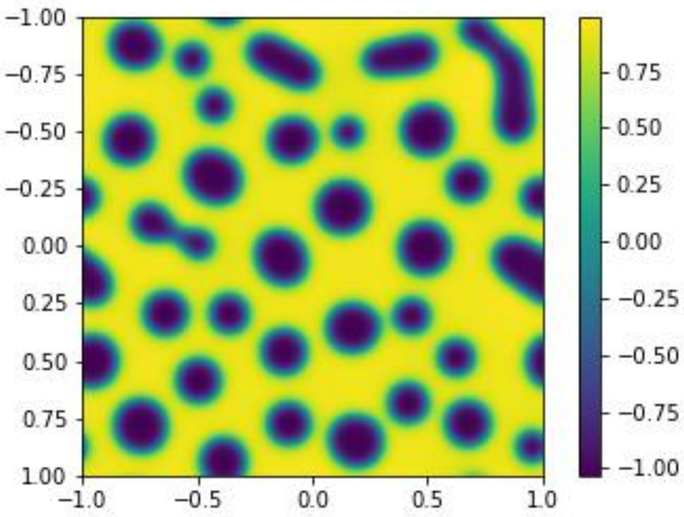
Changing initial composition

One interesting phenomena in binary liquid systems is the formation of droplets from a single phase, these can be formed by adjusting the initial state of the liquid

By adjusting the parameters as shown below and in the next slide, we see droplets form



Initial composition

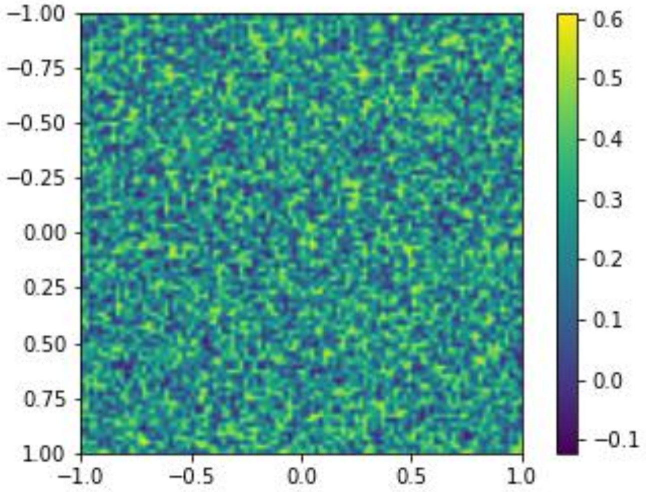


Final composition

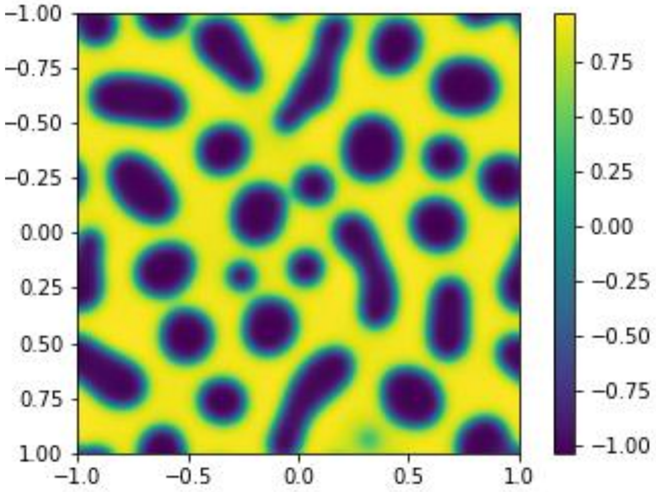
α=10, β=100, initial composition distributed between [-0.1, 0.9]

Droplets of one composition forms, occasionally droplets merge to form an elongated domain

Changing initial composition



Initial composition



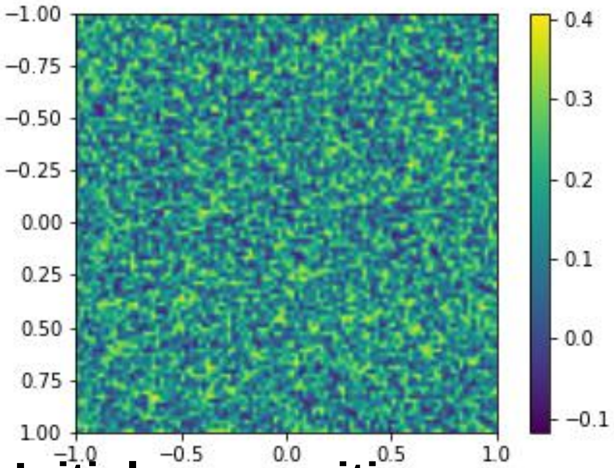
Final composition

α=10, β=100, initial composition distributed between [-0.1, 0.6]

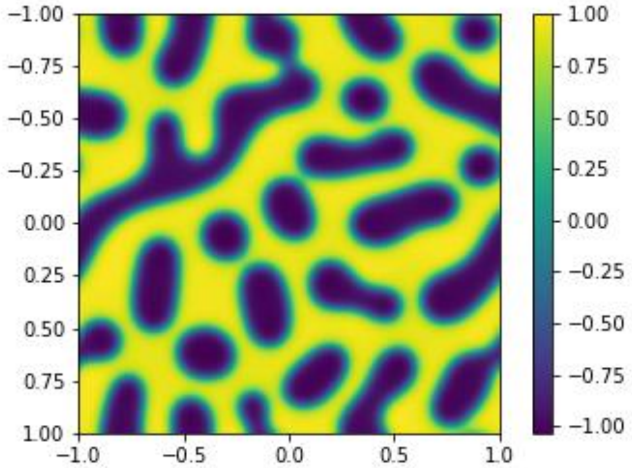
Droplets begin to form, elongated domain reduced in size

α=10, β=100, initial composition distributed between [-0.1, 0.4]

Droplets begin to form, but long domains are still retained



Initial composition



Final composition