Numerically Solving the Allen-Cahn Equation

Josh Arrington

APC 523

May 07, 2024

Allen-Cahn Equation

- Describes the evolution of some *non-conserved* order parameter, ϕ

$$\tau \frac{\partial \phi}{\partial t} = -\frac{\delta F}{\delta \phi}$$

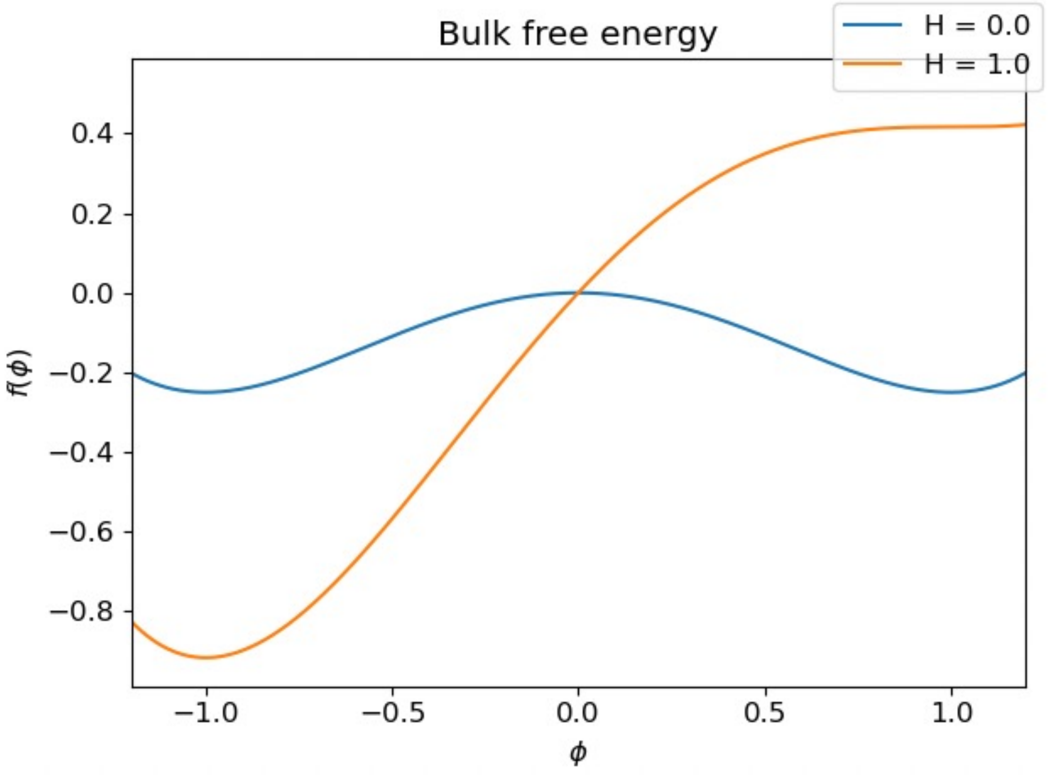
- The free energy functional includes a bulk term and a leading-order interfacial term:

$$F[\phi] = \int_V \left(f(\phi) + \frac{w^2}{2} (\nabla \phi)^2 \right) \mathrm{d}V$$

$$\Rightarrow -\frac{\delta F}{\delta \phi} = -f'(\phi) + w^2 \nabla^2 \phi$$

Double-well bulk free energy

- Double-well bulk free energy allows two stable phases
$$f(\phi) = -\frac{1}{2}\phi^2 + \frac{1}{4}\phi^4 + H\left(\phi - \frac{1}{3}\phi^3\right)$$
- H controls asymmetry of the two minima
- In this case, the Allen-Cahn equation becomes
$$\tau\frac{\partial\phi}{\partial t} = w^2\nabla^2\phi + \phi - \phi^3 - H(1-\phi^2)$$

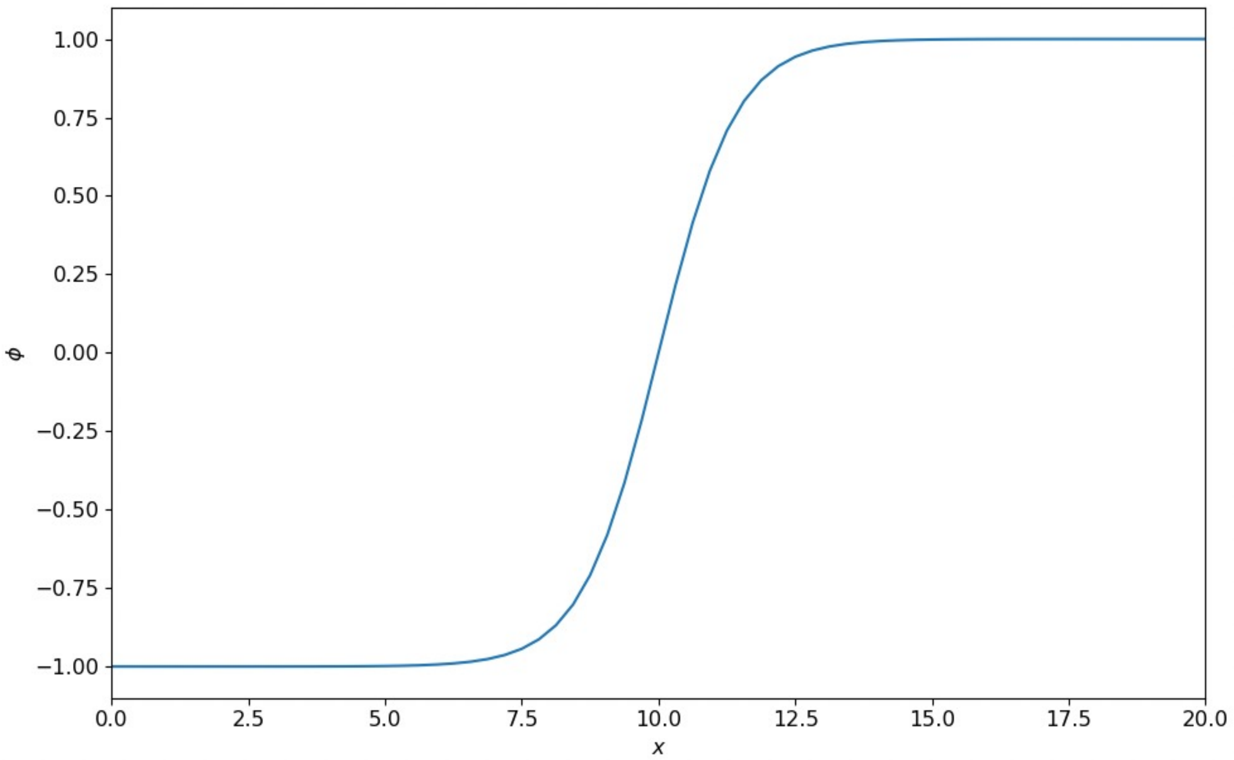


Analytically solving Allen-Cahn equation

$$\tau \frac{\partial \phi}{\partial t} = w^2 \nabla^2 \phi + \phi - \phi^3 - H(1 - \phi^2)$$

- Describes evolution of a pure system with two phases at $\phi = \pm 1$
- Well known equilibrium solution $(H = 0, \phi_t = 0)$ (in 1D, e.g.):

$$\phi_{eq}(x) = \tanh\left(\frac{\mathrm{x}}{\sqrt{2}w}\right)$$



Analytically solving Allen-Cahn equation

- Method of characteristics can help us find non-equilibrium solutions (if H is not large):

$$\phi(x,t) \approx \phi_{eq}(x - vt); u \equiv x - vt$$

- It can be shown that the velocity of this "traveling wave" solution is constant with the form

$$v = \frac{4}{3}\frac{H}{\tau\alpha}; \alpha \equiv \int_{-\infty}^{\infty} \left(\frac{\mathrm{d}\phi_{eq}(x)}{\mathrm{d}x}\right)^2 \mathrm{d}x$$

- Generalizing to 2D with circular domain/interface with $H = 0$:

$$A(t) - A(t = 0) = -\frac{2\pi t}{\tau}$$

Numerical Methods

- I solve the Allen-Cahn equation in 1D and 2D using:
  - Finite difference methods (e.g., Forward Euler)

$$\tau \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = w^2 \frac{-\phi_{i+2}^n + 16\phi_{i+1}^n - 30\phi_i^n + 16\phi_{i-1}^n - \phi_{i-2}^n}{12(\Delta x)^2} - f'(\phi_i^n)$$

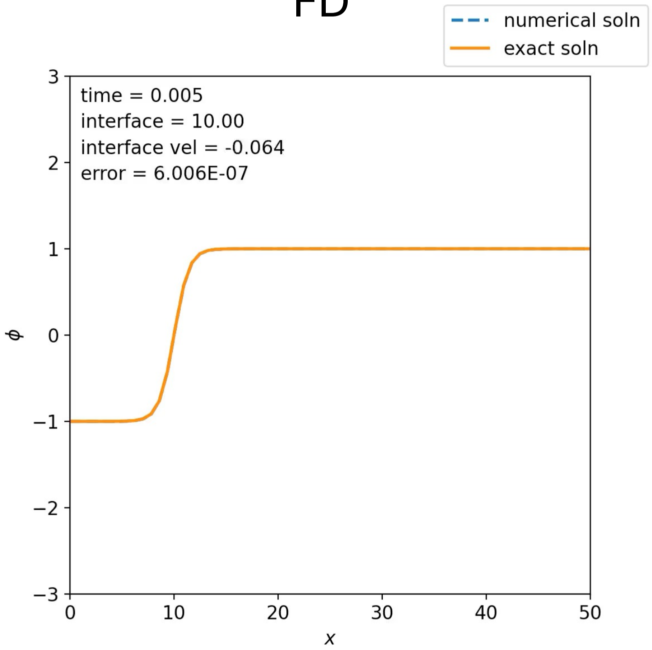
  - RK4 was also used (algorithm not included for brevity)
  - Spectral methods (e.g., semi-implicit Euler)

$$\tau \frac{\{\phi_i\}_k^{n+1} - \{\phi\}_k^n}{\Delta t} = -\{f'(\phi_i)\}_k^n - k^2 w^2 \{\phi_i\}_k^{n+1}$$

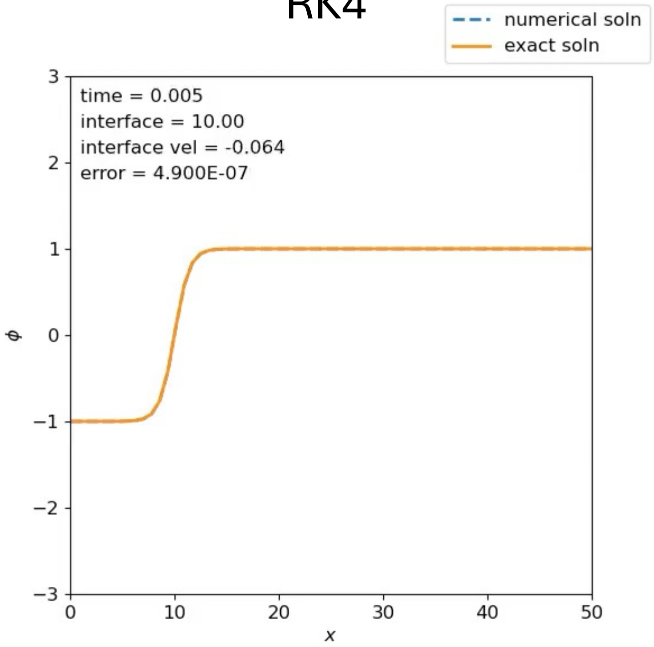
$$\Rightarrow \{\phi_i\}_k^{n+1} = \frac{-\frac{\Delta t}{\tau}\{f'(\phi_i)\}_k^n + \{\phi_i\}_k^n}{1 + \frac{\Delta t}{\tau} w^2 k^2}$$

1D solution: profile evolution

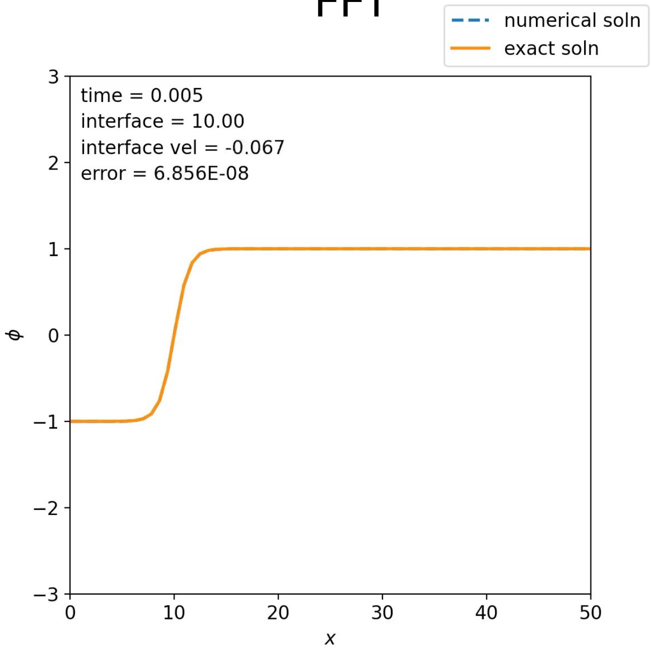
FD



RK4

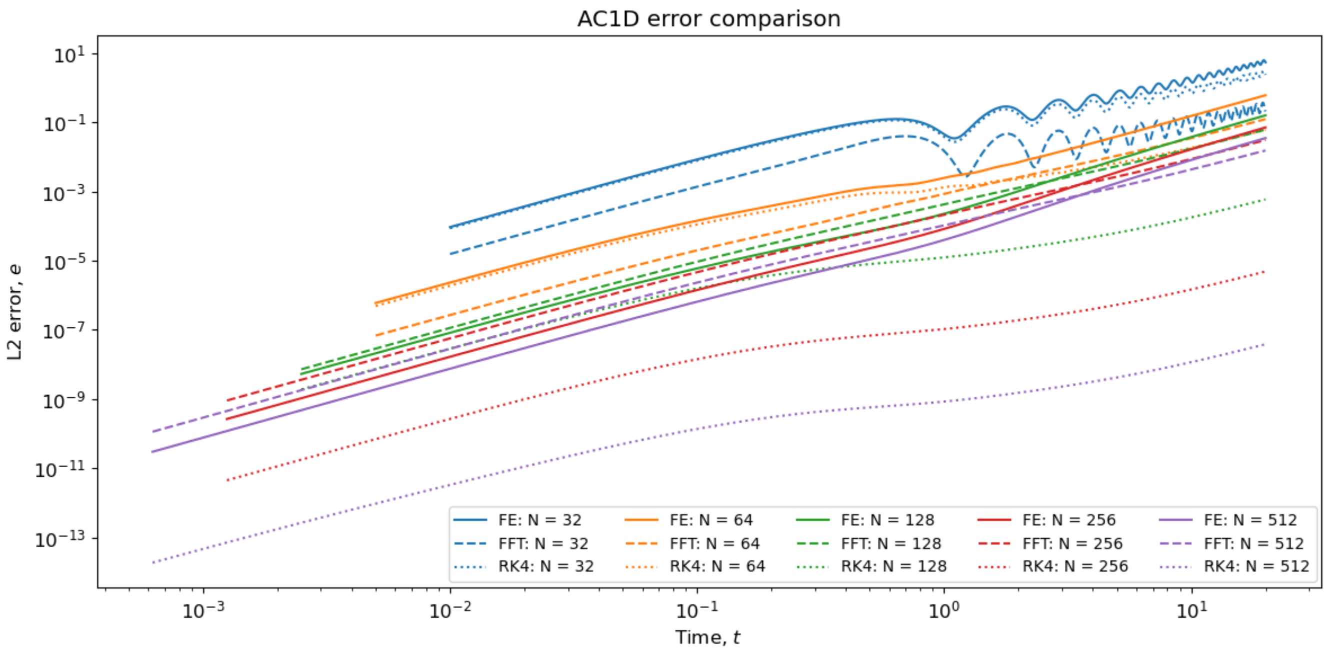


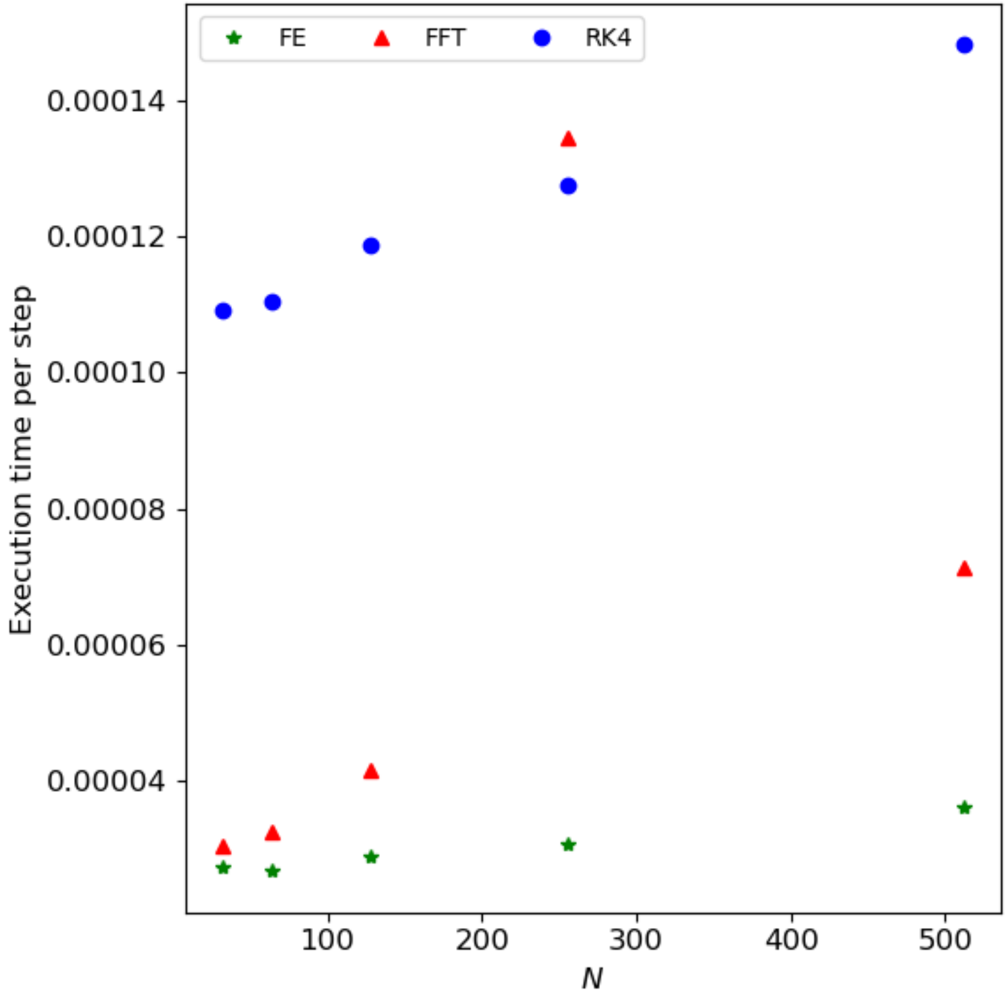
FFT



1D solution: error and execution time

- Same timestep used for all methods
- $H, \tau, w = 1$
- Best performance from RK4 (since higher timestep could be used compared to FE)

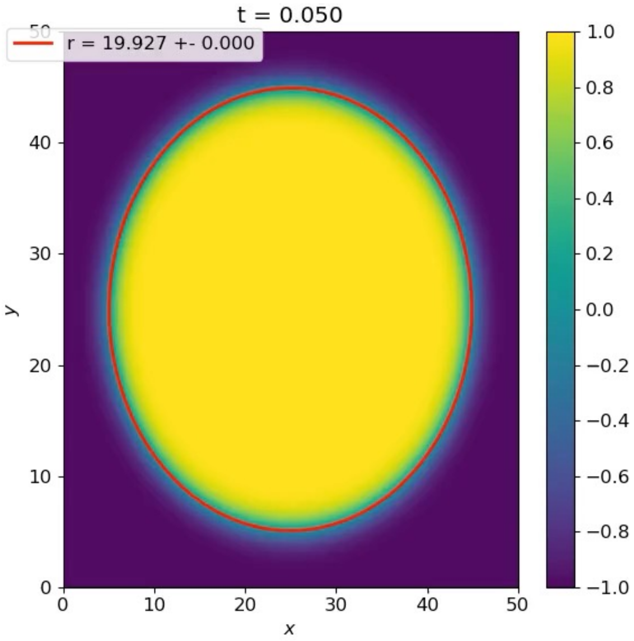




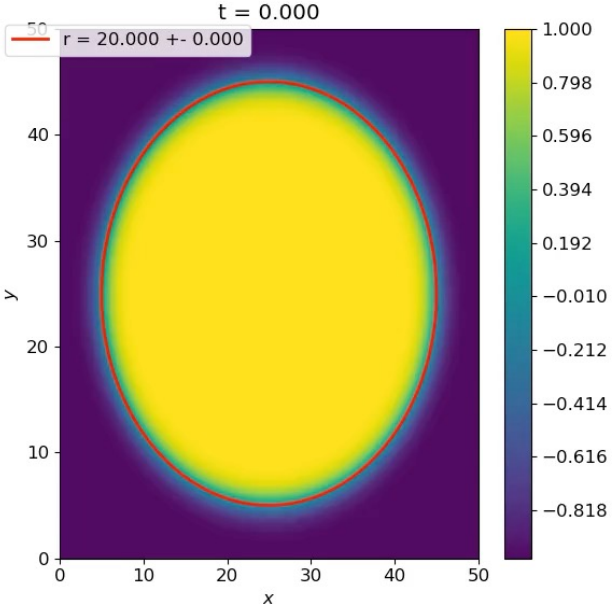
2D solution: evolution of interface

- $H \neq 0$ for illustrative purposes; error compared to analytical evaluated for $H = 0$

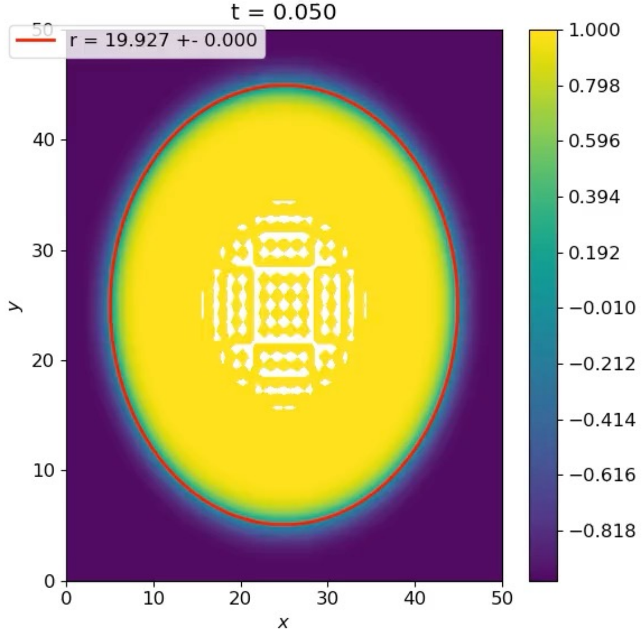
FE



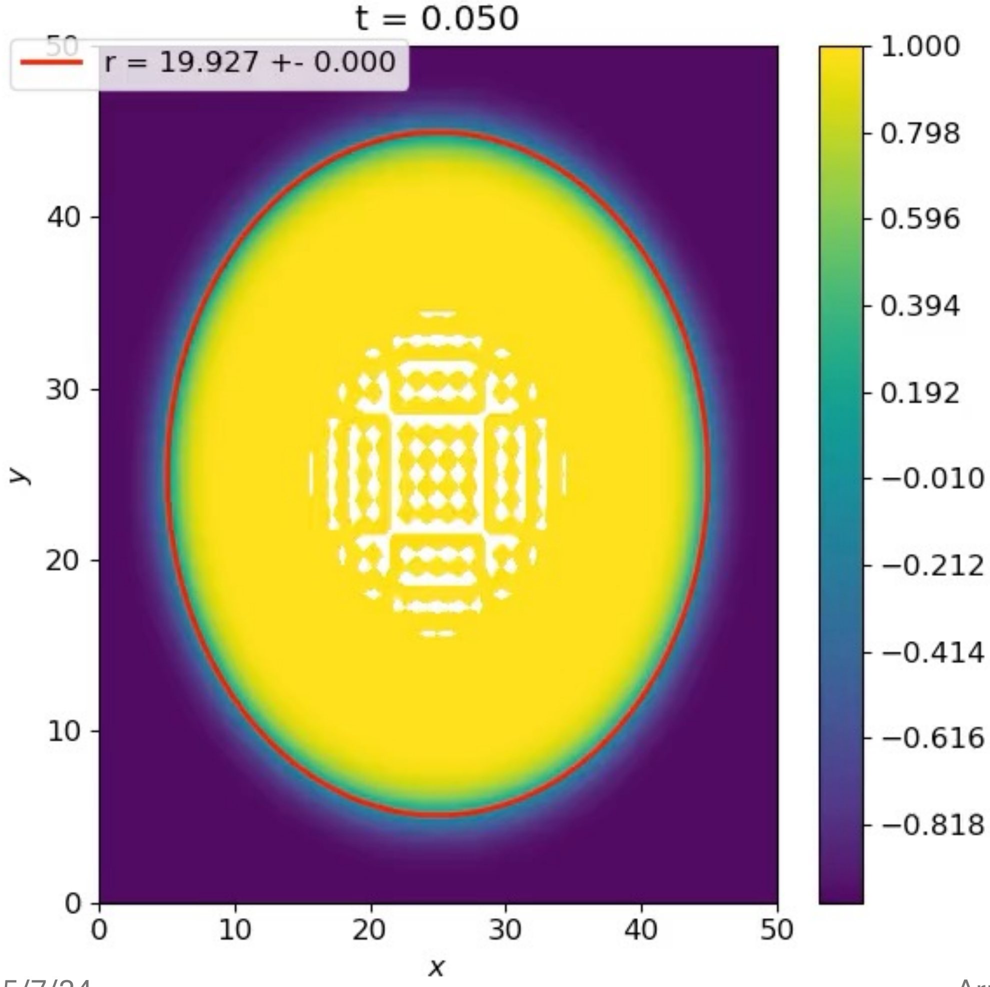
RK4

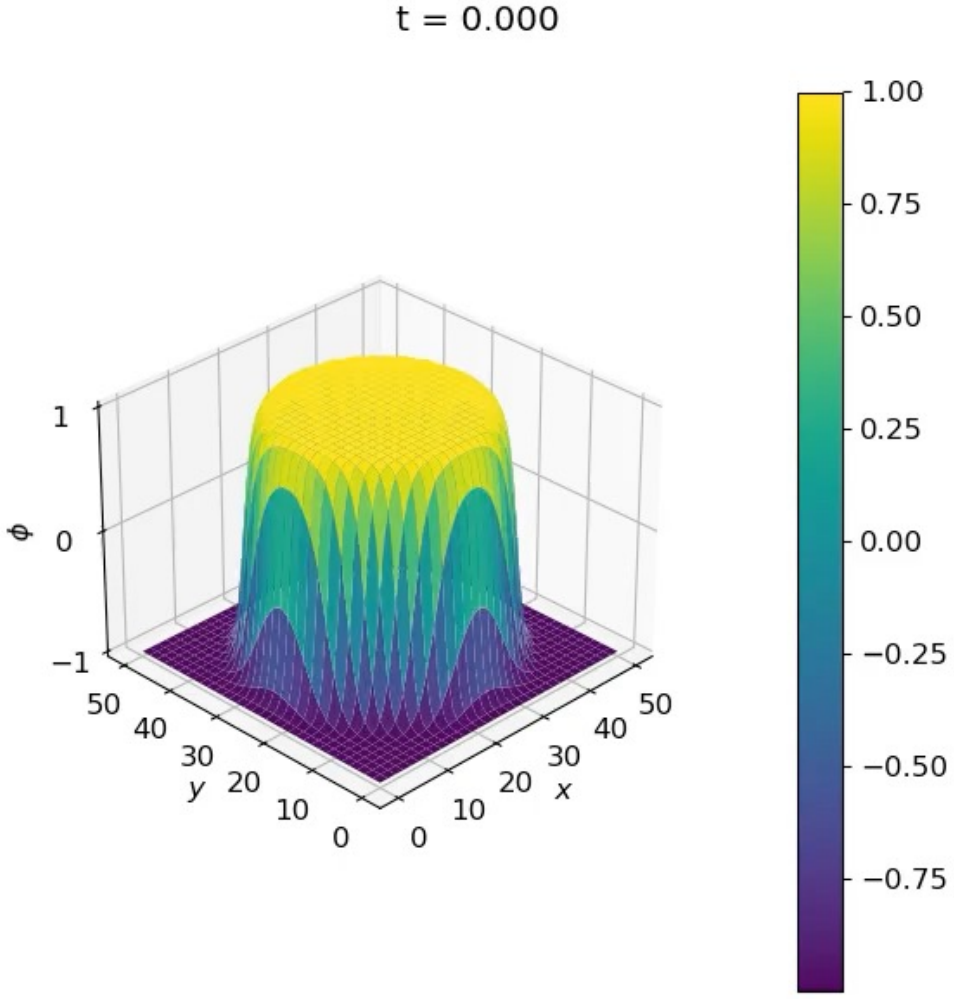


FFT



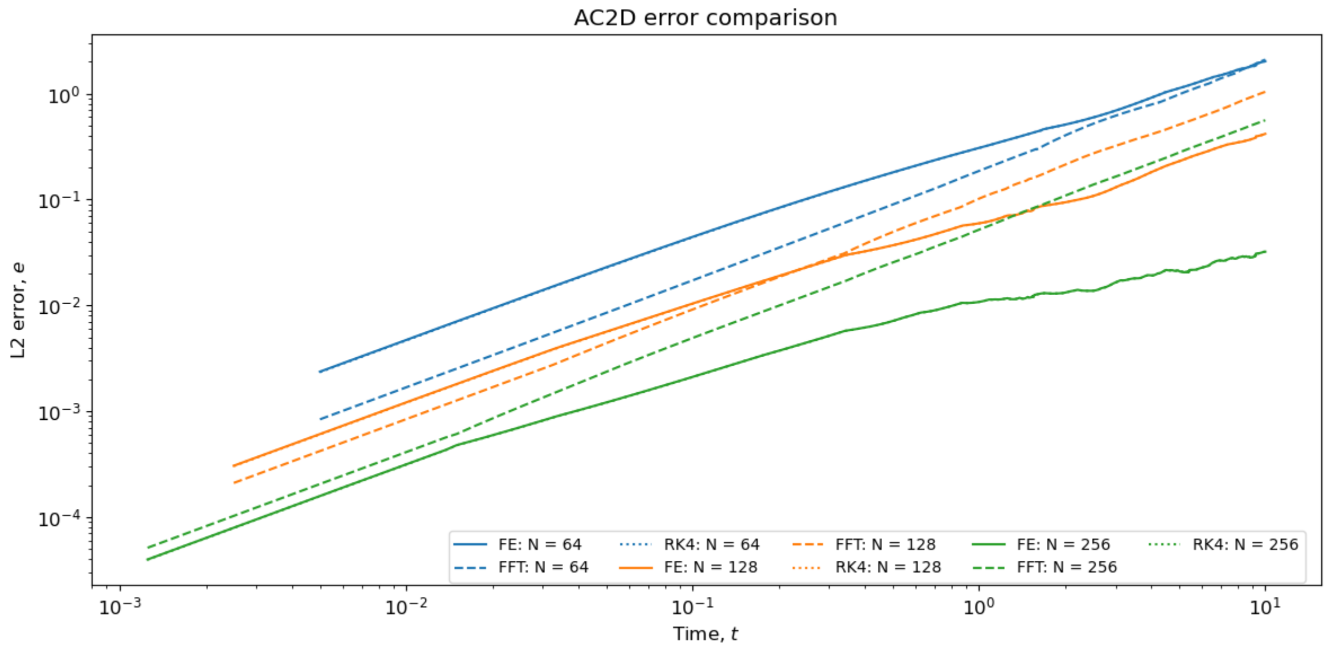
2D FFT solution: oscillations miniscule in magnitude

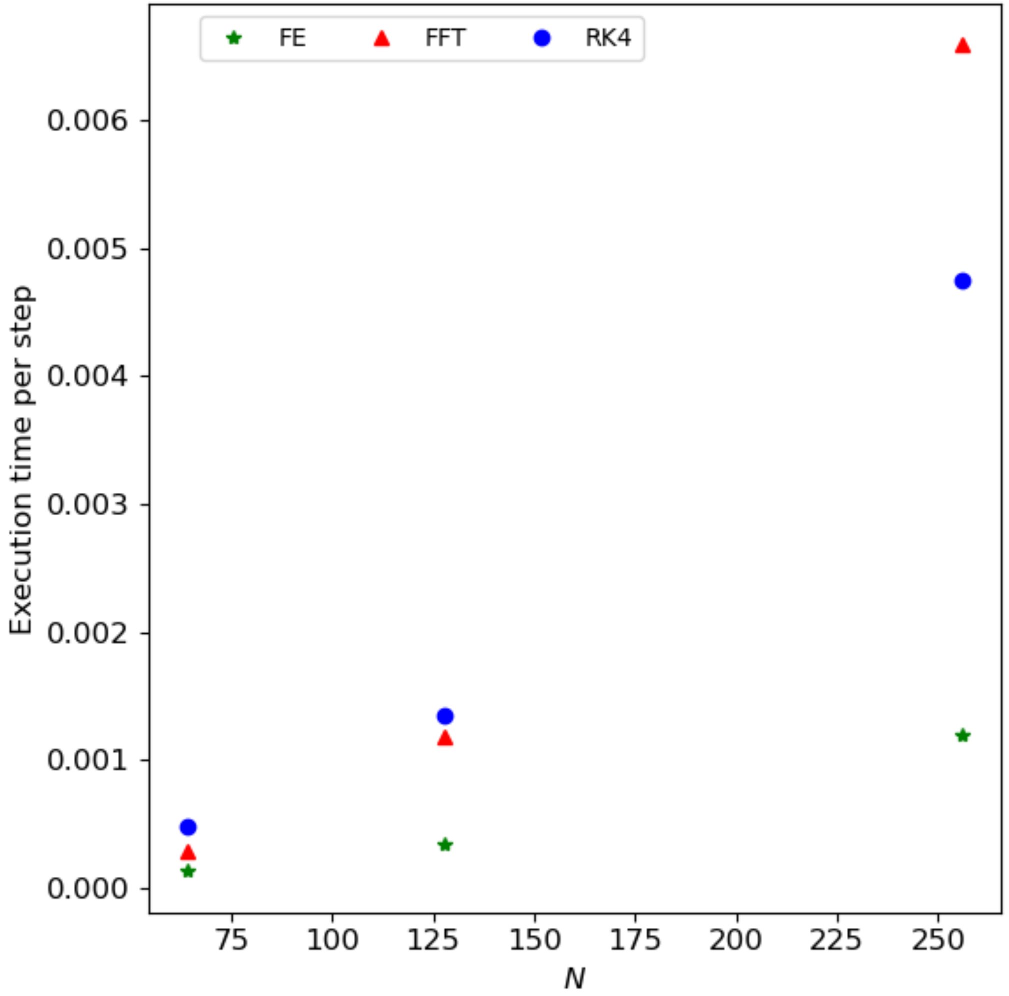




2D solution: error and execution time

- Same timestep used for all methods
- $H = 0,\ \tau, w = 1$
- FE/RK4 similar due to very small driving force for motion in 2D when $H = 0$
- FFT particularly bad in 2D due to need to transform/inverse transform at each step





Conclusions

- Elementary version of Allen-Cahn equation implemented in Python
- In 1D, RK4 is very competitive for simple forms of the free energy functional
- In 2D, FFT becomes less tractable due to need to forward and inverse Fourier transform the ϕ at each timestep
- In practice, multiple components and highly nonlinear forms of the free energy functional lead to multiple coupled nonlinear PDEs describing the evolution of ϕ