

# APC 523 Final Project: Numerically solving the Allen-Cahn equation

Josh Arrington

May 07, 2024

## 1 Introduction and Background

Phase-field modeling involves a class of modeling used to study physical systems involving interfaces between two distinct phases. Here, we use "phase" in the materials science sense, referring to a region in space where the system of interest is homogeneous in material properties (composition, mechanical properties, electromagnetic property, structure, etc.). Material systems in most real-world engineering applications have multiple phases throughout, with the proportion and arrangement of different phases evolving over time.

Consider, for example, a phase-field parameter  $\phi \in [-1, 1]$  in a system with two phases,  $A$  and  $B$ , where  $\phi = -1$  in phase  $A$  and  $\phi = 1$  in phase  $B$ . The historical "sharp interface" approach would view all interfaces between phase  $A$  and phase  $B$  as points where  $\phi$  varies discontinuously, in a step-wise manner, between the two phases. This approach would require different differential equations describing the evolution of some quantity in the two different phases as well as boundary conditions at the interfaces. Solving these coupled equations even for two phases is non-trivial, and often non-tractable, because the interfaces and the relevant quantities of interest in the systems are constantly in flux in most systems and processes of interest. Phase-field modeling seeks to rectify this by allowing the phase-field parameter  $\phi$  to vary smoothly and continuously from one phase to another.

One of the classical phase-field models is the Allen-Cahn equation [1, 2]:

$$\tau \frac{\partial \phi}{\partial t} = - \frac{\delta F[\phi]}{\delta \phi} \quad (1)$$

where  $\phi(\vec{r}, t)$  is a field parameter of species with spatial and temporal variance,  $\tau$  is a control parameter, and  $F$  is a free energy functional of  $\phi$ . The  $\delta$  notation refers to a variational derivative (i.e.,  $\frac{\delta F}{\delta \phi}$  is the variational derivative of  $F$  with respect to  $\phi$ ).

The Allen-Cahn equation is *non*-conserving for the time evolution of  $\phi$ , and is used to model other types of phase transitions like solidification in alloys, grain growth, etc [3]. This equation is a gradient flow of the energy functional  $F$ , and thus requires "energy stability" meaning that  $F$  is never increasing as time evolution is being performed ( $\frac{d}{dt} F[\phi] \leq 0$  for all  $t > 0$ ), which is an important property for numerical methods to conserve.

The focus of this project will be on **the Allen-Cahn Equation, Eq.(1)**, and implementing various numerical methods to solve approximate a solution for an analytical toy problem.

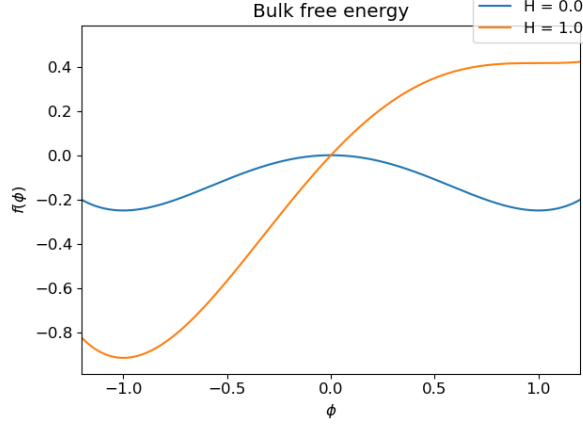


Figure 1: Bulk free energy functional with  $H = 0$  (blue) and  $H = 1$  (orange)

## 2 Analytical solution to the Allen-Cahn equation

Following the literature [4], I derive an analytical solution to the Allen-Cahn equation. In this work, I consider a pure system with one component whose order is represented by  $\phi$  with two phases at  $\phi = \pm 1$ . To accomodate diffuse interfaces, the free energy functional considered here is of the form

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

which consists of a bulk free energy functional,  $f(\phi)$ , and the lowest-order term to account for local variations [5] in the phase field. The parameter  $w$  is representative of the mobility of the species. To promote a two-phase solution, a so-called "double well" bulk free energy functional is considered:

$$f(\phi) = -\frac{1}{2}\phi^2 + \frac{1}{4}\phi^4 + H(\phi - \frac{1}{3}\phi^3) \quad (3)$$

with minima at  $\phi = \pm 1$ .  $H$  controls the asymmetry of the two minima in the region  $\phi \in [-1, 1]$ . This role of  $H$  is seen in Figure 1, where a positive value skews the minima at  $\phi = -1$  lower than the minima at  $\phi = 0$ .

With this chose form of our free energy functional, then upon taking the variational derivative of Eq. (2) the Allen-Cahn equation becomes

$$\tau \frac{\partial \phi}{\partial t} = w^2 \nabla^2 \phi + \phi - \phi^3 - H(1 - \phi^2) \quad (4)$$

### 2.1 1D Allen-Cahn equation

Let us consider the 1D case first, where  $x$  denotes our spatial variable extending from  $-\infty$  to  $\infty$ . At steady state and when  $H = 0$ ,  $\frac{\partial \phi}{\partial t} \rightarrow 0$  and Eq. (4) becomes

$$w^2 \frac{d^2 \phi}{dx^2} + \phi - \phi^3 = 0 \quad (5)$$

Then an equilibrium solution to Eq. (4) in 1D is clearly:

$$\phi_{eq}(x) = \tanh \left( \frac{x}{\sqrt{2}w} \right) \quad (6)$$

We can verify this:

$$\begin{aligned} \frac{d^2\phi_{eq}}{dx^2} &= -\frac{1}{w^2} \left( \tanh\left(\frac{x}{\sqrt{2}w}\right) - \tanh^3\left(\frac{x}{\sqrt{2}w}\right) \right) \\ \Rightarrow -\frac{w^2}{w^2} \left( \tanh\left(\frac{x}{\sqrt{2}w}\right) - \tanh^3\left(\frac{x}{\sqrt{2}w}\right) \right) + \phi_{eq} - \phi_{eq}^3 &= 0 \end{aligned}$$

We can interpret this 1D solution physically as a smooth transition from  $\phi = -1$  to  $\phi = 1$ , and we can define an interface as the point where  $\phi = 0$  (for the purpose of tracking the interface's motion). Thus, with an initial state of the form of Eq. (5), then  $\frac{\partial\phi}{\partial t} = 0$  and the system would be stagnant (the interface). However, if  $H \neq 0$ , then we would have a driving force for the motion of the interface between the two phases and  $\frac{\partial\phi}{\partial t} \neq 0$ .

Let us consider  $H > 0$  but not too large. It has been shown in the literature that such a scenario results in a solution that the interface propagating at constant speed  $v$ . Following [4], we can recover this result through the method of characteristics, similar to the advection equation. We approximate  $\phi(x, t) \approx \phi_{eq}(x - vt) = \phi(u)$  and define  $u \equiv x - vt$ . Then, the 1D Allen-Cahn equation becomes

$$\tau \frac{d\phi_{eq}}{du} \frac{\partial u}{\partial t} = -v\tau \frac{d\phi_{eq}}{du} = w^2 \frac{d^2\phi_{eq}}{du^2} + \phi_{eq} - \phi_{eq}^3 - H(1 - \phi_{eq}^2) \quad (7)$$

However, by construction, the first three terms on the right-hand side of Eq. (7) combine to 0 (since  $\phi_{eq}$  is a solution to Eq. (5)). Thus, we can write

$$-v\tau \frac{d\phi_{eq}}{du} = -H(1 - \phi_{eq}^2) \quad (8)$$

To derive what the constant velocity of the interface  $v$  is, we use the trick of integrating over both sides of Eq. (8) with  $\frac{d\phi_{eq}}{du}$  as our "kernel":

$$\begin{aligned} \int_{-\infty}^{\infty} \left( \frac{d\phi_{eq}}{du} \left( -v\tau \frac{d\phi_{eq}}{du} \right) \right) du &= \int_{-\infty}^{\infty} \left( -H(1 - \phi_{eq}^2) \right) du \\ -v\tau \int_{-\infty}^{\infty} \left( \frac{d\phi_{eq}}{du} \right)^2 du &= -H \int_{-\infty}^{\infty} \frac{d}{du} \left( \phi_{eq} - \frac{1}{3}\phi_{eq}^3 \right) du \\ -v\tau \int_{-\infty}^{\infty} \left( \frac{d\phi_{eq}}{du} \right)^2 du &= -H \left( \phi_{eq} - \frac{1}{3}\phi_{eq}^3 \right) \Big|_{-\infty}^{\infty} \\ \Rightarrow v &= \frac{4H}{3\tau} \left( \int_{-\infty}^{\infty} \left( \frac{d\phi_{eq}}{du} \right)^2 du \right)^{-1} \end{aligned} \quad (9)$$

We see that the velocity is indeed constant and proportional to the controlling parameters  $H, \tau, w$ .

## 2.2 2D Allen-Cahn equation

Now, consider a perfectly circular domain in 2D with  $\phi = 1$  inside the domain, and  $\phi = -1$  outside the domain. Following the above analysis and again utilizing the method of characteristics, we assume that  $\phi(r, t) \approx \phi_{eq}(r - R(t))$  where  $r$  is the initial domain

radius and  $R(t)$  is the radius of the domain at time  $t$ . Taking the laplacian in polar coordinates, we are led to the 2D Allen-Cahn equation for this scenario as

$$\begin{aligned} -\tau \frac{d\phi_{eq}}{du} \frac{dR}{dt} &= w^2 \frac{d^2\phi_{eq}}{du^2} + \phi_{eq} - \phi_{eq}^3 - H(1 - \phi_{eq}^2) + \frac{1}{r} \frac{d\phi_{eq}}{du} \\ \implies -\tau \frac{d\phi_{eq}}{du} \frac{dR}{dt} &= -H(1 - \phi_{eq}^2) + \frac{1}{r} \frac{d\phi_{eq}}{du} \end{aligned} \quad (10)$$

where again the first three terms on the right-hand side sum to zero by construction of  $\phi_{eq}$ .

Again, integrating over both sides with the kernel  $\frac{d\phi_{eq}}{du}$ :

$$\begin{aligned} -\int_{-\infty}^{\infty} \left( \tau \frac{dR}{dt} \left( \frac{d\phi_{eq}}{du} \right)^2 \right) du &= \frac{1}{r} \int_{-\infty}^{\infty} \left( \frac{d\phi_{eq}}{du} \right)^2 du - H \int_{-\infty}^{\infty} \frac{d}{du} \left( \phi_{eq} - \frac{1}{3} \phi_{eq}^3 \right) du \\ -\tau \alpha \frac{dR(t)}{dt} &= \frac{\alpha}{r} - \frac{4H}{3} \\ \implies \frac{dR(t)}{dt} &= -\frac{1}{\tau R(t)} + \frac{4H}{3\tau \alpha} \end{aligned} \quad (11)$$

where in the second line we defined  $\alpha \equiv \int_{-\infty}^{\infty} \left( \frac{d\phi_{eq}}{du} \right)^2 du$ .

Eq. (11) holds for all  $H \neq 0$  that are not too large in magnitude. For further analysis, we consider the case when  $H = 0$ . Then, Eq. (11) becomes

$$\begin{aligned} \frac{dR(t)}{dt} &= -\frac{1}{\tau R(t)} \\ R(t)^2 - R(0)^2 &= -\frac{2t}{\tau} \\ \implies A(t) &= A(0) - \frac{2t}{\tau} \end{aligned} \quad (12)$$

where in the second line we solved the ODE by separation of variables and in the third line we multiplied both sides by  $\pi$  to get the area of the circular domain. Thus, Eq. (12) predicts that even for a symmetric bulk free energy ( $H = 0$ ), the circular domain would shrink for  $\tau > 0$ . The physical interpretation for this is that surface tension results in an energetic cost for a curved interface in 2D.

### 3 Numerical Algorithms

For both the 1D and 2D cases, I implement the following numerical algorithms:

- Finite difference solver with explicit Forward Euler (FE) time-stepping
- Finite difference solver with fourth-order Runge-Kutta (RK4) time-stepping
- Spectral methods with semi-implicit methods

This was done in Python 3.12.2 with the standard scientific libraries of Numpy, Matplotlib, and Scipy.

### 3.1 1D numerical algorithms

In 1D, space is discretized as  $x_i \in [0, L]$ ,  $i = 0, \dots, N$ . The spatial resolution is then  $\Delta x = L/N$ . For our 1D case, the finite-difference FE algorithm is quite straight forward. We approximate the second derivative  $\frac{d^2\phi}{dx^2}$  using the fourth-order, five-point central finite difference stencil:

$$\frac{d^2\phi(x_i^n)}{dx^2} \approx \frac{-\phi_{i+2}^n + 16\phi_{i+1}^n - 30\phi_i^n + 16\phi_{i-1}^n - \phi_{i-2}^n}{12(\Delta x)^2} \quad (13)$$

For  $\phi_i$  near the boundary, the boundary values are extended to imitate a semi-infinite domain. Then the explicit FE solver for the 1D Eq. (4) is simply

$$\tau \frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} = w^2 \frac{-\phi_{i+2}^n + 16\phi_{i+1}^n - 30\phi_i^n + 16\phi_{i-1}^n - \phi_{i-2}^n}{12(\Delta x)^2} - f'(\phi_i^n) \quad (14)$$

$$-f'(\phi_i^n) = \phi_i^n - (\phi_i^n)^3 - H(1 - (\phi_i^n)^2) \quad (15)$$

However, we know that FE is both restrictive on the timestep, as well as first-order accurate. From Lecture 13, we know that RK4 has a wider stability region (and can thus make larger timesteps) and is also fourth-order accurate. The RK4 algorithm for the 1D Allen-Cahn equation then becomes:

$$\begin{aligned} g(\phi_i^n) &\equiv \frac{1}{\tau} \left( w^2 \frac{-\phi_{i+2}^n + 16\phi_{i+1}^n - 30\phi_i^n + 16\phi_{i-1}^n - \phi_{i-2}^n}{12(\Delta x)^2} + f'(\phi_i^n) \right) \\ \phi_1 &= \phi^n, \quad k_1 = g(\phi_1) \\ \phi_2 &= \phi^n + \frac{\Delta t}{2} k_1, \quad k_2 = g(\phi_2) \\ \phi_3 &= \phi^n + \frac{\Delta t}{2} k_2, \quad k_3 = g(\phi_3) \\ \phi_4 &= \phi^n + k_3 \Delta t, \quad k_4 = g(\phi_4) \\ \phi^{n+1} &= \phi^n + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4) \end{aligned} \quad (16)$$

Note that the numbered subscripts in the stages refer to the global *stage* and not the discrete value of  $\phi$  at that index.

Finally, the semi-implicit spectral method (following [3]) treats the linear and second-order terms implicitly and the nonlinear  $f'(\phi)$  terms explicitly. We denote the Fourier transform of  $\phi$  as  $\{\phi\}_k$  and recognize that

$$\left\{ \frac{\partial^n \phi}{\partial x^n} \right\}_k = ((\sqrt{-1})^n k)^n \{\phi\}_k$$

Then, the semi-implicit spectral form of Eq. (4)

$$\begin{aligned} \tau \frac{\{\phi_i\}_k^{n+1} - \{\phi_i\}_k^n}{\Delta t} &= -\{f'(\phi_i)\}_k^n - k^2 w^2 \{\phi_i\}_k^{n+1} \\ \implies \{\phi_i\}_k^{n+1} &= \frac{\{\phi_i\}_k^n - \frac{\Delta t}{\tau} \{f'(\phi_i)\}_k^n}{1 + \frac{\Delta t}{\tau} k^2 w^2} \end{aligned} \quad (17)$$

To enforce periodic boundary conditions in the 1D system, the domain of simulation is mirrored across the right boundary. It is important to note that due to the nonlinearity of  $f'(\phi)$ , it is evaluated in real space at each time step. This means that at each timestep, the spectral method requires:

1. Fourier transform  $\phi_i^n$  and  $f'(\phi_i^n)$ .
2. Evaluate Eq. (17)
3. Inverse Fourier transform  $\{\phi_i\}_k^{n+1}$

This can become expensive as systems reach large  $N$ .

### 3.2 2D numerical algorithms

In 2D, space is discretized as  $(x_i, y_j) \in [0, L] \times [0, L]$ ,  $i, j = 0, \dots, N$ . This is a square domain and the spatial resolution is  $\Delta x = \Delta y = L/N \equiv h$ . The Laplacian is approximated using the second-order 5-point central difference approximation:

$$\nabla^2 \phi_{i,j}^n \approx \frac{\phi_{i+1,j}^n + \phi_{i-1,j}^n + \phi_{i,j+1}^n + \phi_{i,j-1}^n - 4\phi_{i,j}^n}{h^2} \quad (18)$$

Again, for  $\phi_{i,j}$  near the boundary, the boundary values are extended to imitate a semi-infinite domain. Then the explicit FE solver for the 2D Eq. (4) is simply

$$\tau \frac{\phi_{i,j}^{n+1} - \phi_{i,j}^n}{\Delta t} = w^2 \frac{\phi_{i+1,j}^n + \phi_{i-1,j}^n + \phi_{i,j+1}^n + \phi_{i,j-1}^n - 4\phi_{i,j}^n}{h^2} - f'(\phi_{i,j}^n) \quad (19)$$

$$-f'(\phi_{i,j}^n) = \phi_{i,j}^n - (\phi_{i,j}^n)^3 - H(1 - (\phi_{i,j}^n)^2) \quad (20)$$

Similarly, the RK4 algorithm for the 2D Allen-Cahn equation then becomes:

$$\begin{aligned} g(\phi_{i,j}^n) &\equiv \frac{1}{\tau} \left( w^2 \frac{\phi_{i+1,j}^n + \phi_{i-1,j}^n + \phi_{i,j+1}^n + \phi_{i,j-1}^n - 4\phi_{i,j}^n}{h^2} - f'(\phi_{i,j}^n) \right) \\ \phi_1 &= \phi^n, \quad k_1 = g(\phi_1) \\ \phi_2 &= \phi^n + \frac{\Delta t}{2} k_1, \quad k_2 = g(\phi_2) \\ \phi_3 &= \phi^n + \frac{\Delta t}{2} k_2, \quad k_3 = g(\phi_3) \\ \phi_4 &= \phi^n + k_3 \Delta t, \quad k_4 = g(\phi_4) \\ \phi^{n+1} &= \phi^n + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4) \end{aligned} \quad (21)$$

Again, note that the numbered subscripts in each stage of the RK4 algorithm refer to the global stage and not the discrete value of  $\phi$  at that index.

Finally, the semi-implicit spectral method in 2D for the Allen-Cahn equation is trivially similar to Eq. (17):

$$\{\phi_{i,j}\}_k^{n+1} = \frac{\{\phi_{i,j}\}_k^n - \frac{\Delta t}{\tau} \{f'(\phi_{i,j})\}_k^n}{1 + \frac{\Delta t}{\tau} k^2 w^2} \quad (22)$$

For the 2D simulations, the domain involves a circular region of  $\phi = 1$  in the center and  $\phi = -1$  outside of this circle (with a smooth, diffuse transition between them) and so the artificial periodicity needed for the spectral method is naturally accounted for since the domain is symmetric in the radial direction. However, since we do not mirror the domain as in the 1D case, then in 2D we have half of the wavevectors in Fourier space for the same spatial resolution compared to the 1D case (although the Nyquist frequency is the same). The magnitude of the wavevectors in 2D are given as

$$k^2 = k_x^2 + k_y^2 \quad (23)$$

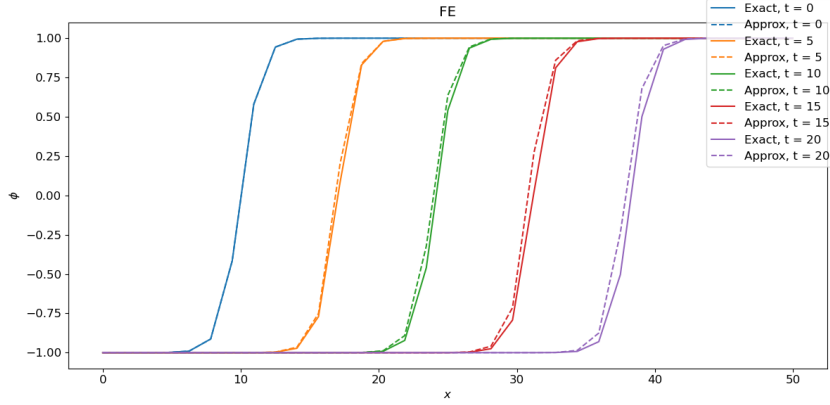


Figure 2: Evolution of profile with FE algorithm

## 4 Results and Discussion

For both the 1D and 2D case, the parameters  $\tau, w, H$  were all set to unity unless otherwise specified. The spatial resolution is  $h = L/N$  and the temporal resolution is set to  $\Delta t \approx h/100$ . Although the RK4 and spectral methods can handle temporal resolution greater than this, it was chosen for stability of the FE algorithm and for consistent comparison of the accuracy of all three methods.

### 4.1 1D results

In 1D,  $N = [32, 64, 128, 256, 512]$ . For all 1D cases, we compare numerical profiles of  $\phi$  versus the known analytical solution discussed in Sec. 2.1:

$$\phi(x, t) = \phi_{eq}(x - vt) = \tanh\left(\frac{x - vt}{\sqrt{2}w}\right) \quad (24)$$

where  $v$  is analytically calculated according to Eq. (9). Evolutions of the profile using FE, RK4, and FFT/spectral methods are shown in Figs. 2, 3, and 4 respectively for  $N = 32$ .

The accuracy of a method is calculated by comparing the profile from the numerical evolution to the profile analytically calculated from the theory using the  $L^2$  norm:

$$e_{L^2}(t) = \int_0^L |\phi_{\text{approx}}(x, t) - \phi_{\text{exact}}(x, t)| dx \quad (25)$$

In Figs. 5 through 7 we see the individual error versus time plots for the three algorithms with varying number of spatial grid points. For direct comparison, all three algorithms' error versus time are plotted on Fig. 8. For low  $N$ , the spectral method has superior accuracy due to the exponential convergence of the spectral methods. However, for large  $N$ , the fourth-order nature of RK4 allows it to become the most accurate.

As discussed above, RK4 is a multistage method and the spectral method requires both a forward and an inverse Fourier transform (which is  $\mathcal{O}(N \log N)$ ). Thus, we compare the execution time for each method in Fig. 9. Clearly, RK4 is the most expensive computationally since it has 4 stages per timestep. However, it is the most accurate method and thus would be preferred for long time simulations.

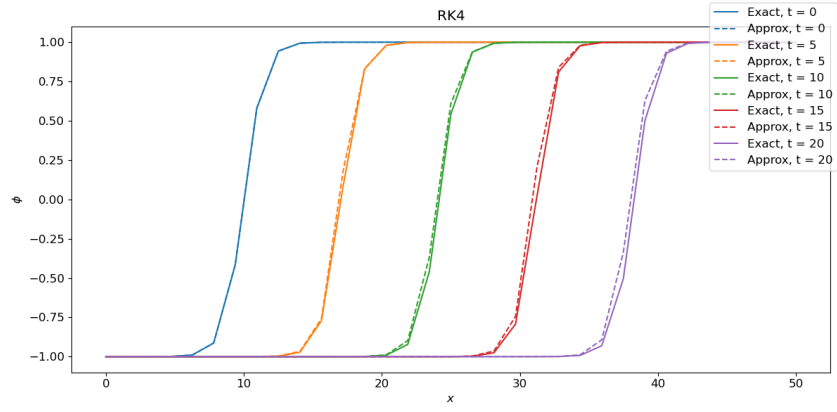


Figure 3: Evolution of profile with RK4 algorithm

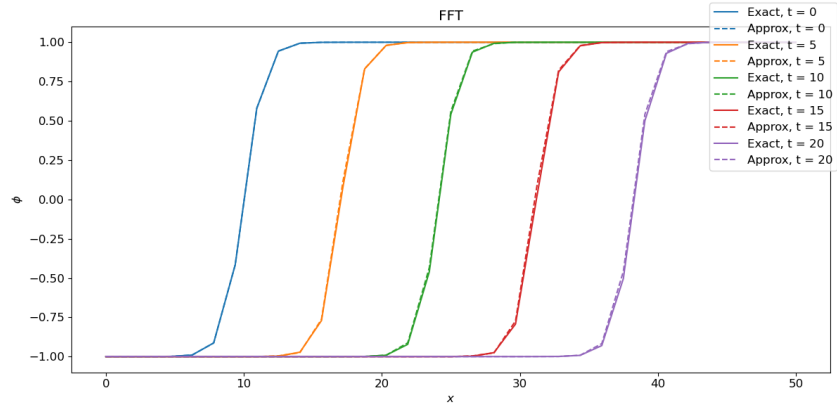


Figure 4: Evolution of profile with FFT algorithm

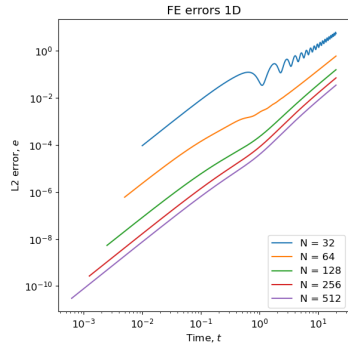


Figure 5: 1D FE error vs time

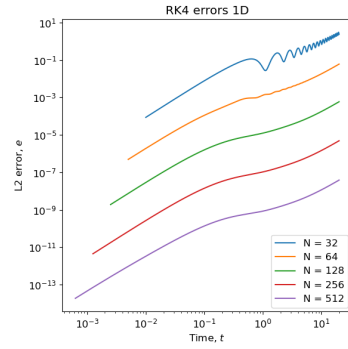


Figure 6: 1D RK4 error vs time

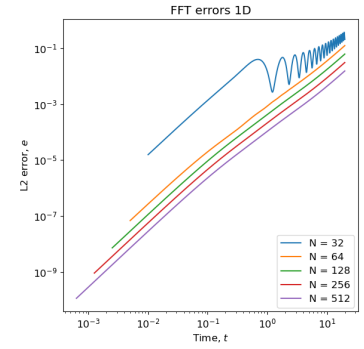


Figure 7: 1D FFT error vs time



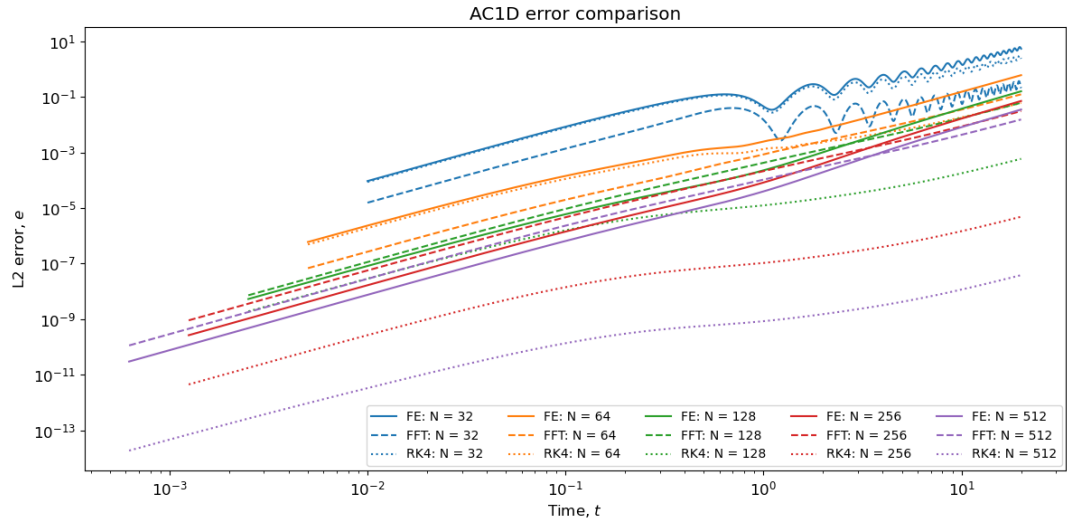


Figure 8: 1D error vs time for the three numerical algorithms

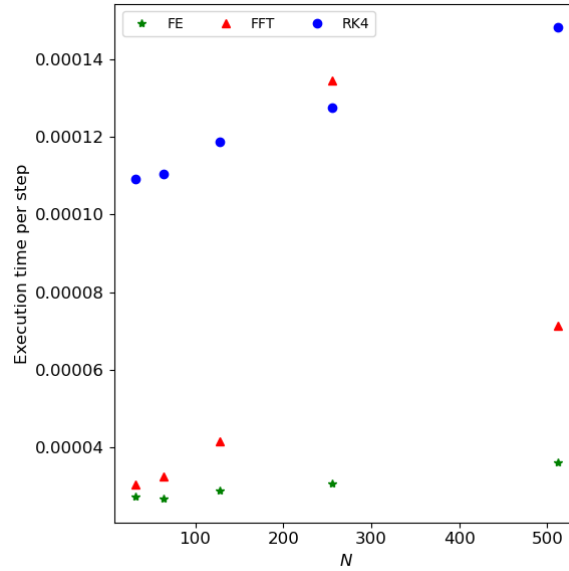


Figure 9: Execution times of the various 1D numerical methods

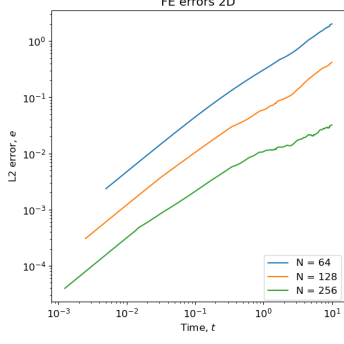


Figure 10: 2D FE error vs time

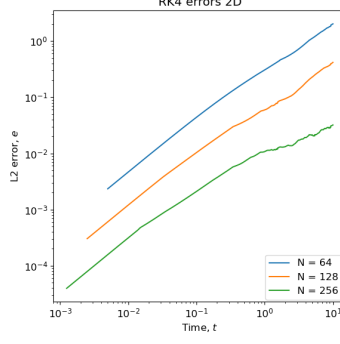


Figure 11: 2D RK4 error vs time

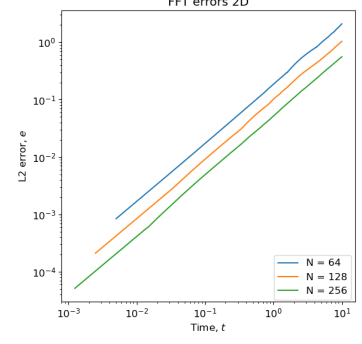


Figure 12: 2D FFT error vs time

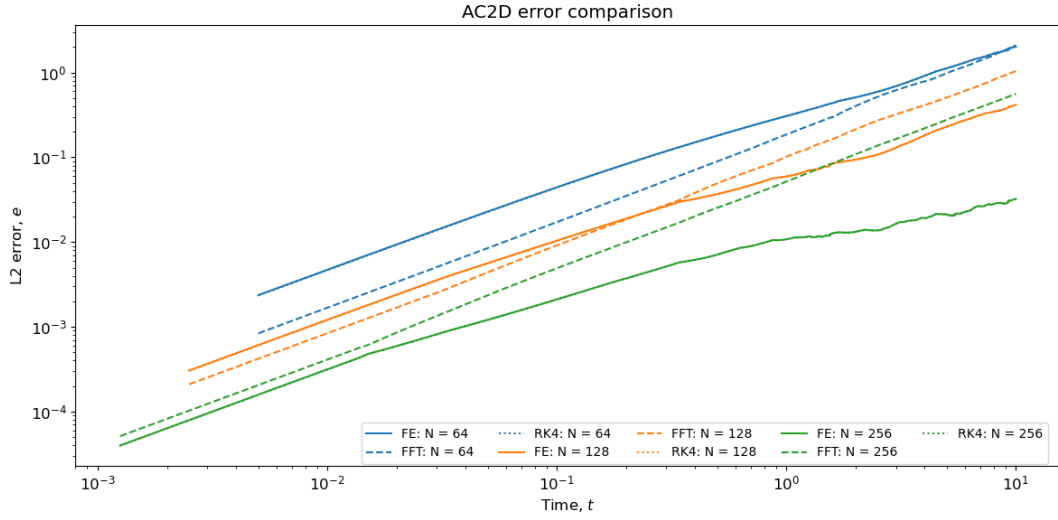


Figure 13: 2D error vs time for the three numerical algorithms

## 4.2 2D results

In 2D,  $N = [64, 128, 256]$  for computational speed for demonstration. For the 2D cases, we set  $H = 0$  for direct comparison against Eq. (12). The initial state is given as

$$\phi(r, 0) = -\tanh\left(\frac{r - R_0}{\sqrt{2}w}\right) \quad (26)$$

(Note: videos of  $H = 1.0$  are shown in the slides/Github).

To analyze the accuracy of the 2D algorithms, the radius of the circle where  $\phi = 0$  is found and then used to compute an approximate area,  $A_{approx} = \pi(r_{\phi=0})^2$ . This is directly compared to  $A_{exact}$  as calculated via Eq. (12):

$$e_{max}(t) = |A_{approx}(t) - A_{exact}(t)| \quad (27)$$

In Figs. 10 through 12 we see the individual error versus time plots for the three algorithms with varying number of spatial grid points. For direct comparison, all three algorithms' error versus time are plotted on Fig. 13. Despite the errors between the two explicit algorithms looks very similar, they are not identical (as verified in Python

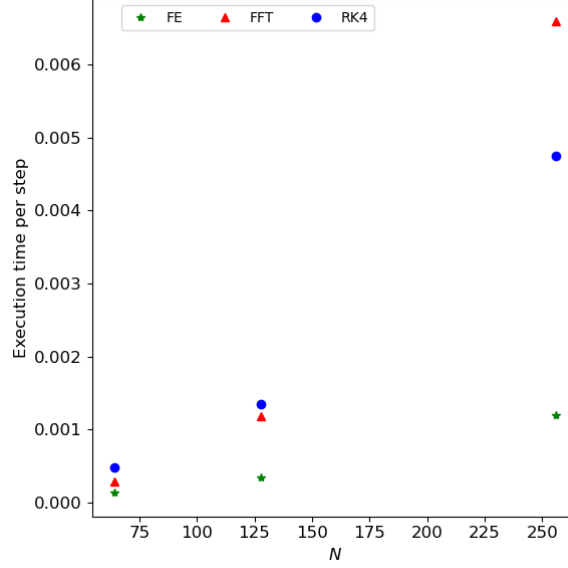


Figure 14: Execution times of the various 2D numerical methods

by comparing the arrays of the datapoints using Numpy's "isclose" function). However, the driving force for interface shrinking in the case when  $H = 0$  is small, and thus the change in area is small and so the slight errors of the different order methods might be comparable. Additionally, we do not see as strong of performance from the spectral method as in the 1D case because, as mentioned above, we are not artificially extending the region of computation and so the spectral method has half the number of wavevectors available.

The execution speeds of the three algorithms are shown in Fig. 14. As stated above, the spectral method requires forward and inverse Fourier transforms computed in both  $x$  and  $y$  directions at each step, which are both  $\mathcal{O}(N^2 \log N)$  operations. As such, the spectral methods become very expensive in dimensions higher than  $D = 1$ , and are not as tractable. For the cases where  $H \neq 0$ , the added accuracy of RK4 would be beneficial for long time simulations, particularly since larger timesteps can be taken compared to the FE that overcome the computational expense.

## 5 Conclusions

In this work, I implemented several numerical algorithms for solving the Allen-Cahn equation in 1D and 2D. The numerical algorithms used were the explicit finite difference forward Euler and fourth-order Runge-Kutta methods, as well as the semi-implicit spectral methods. These methods were implemented in Python and compared against an analytically tractable toy problem that allowed for error reporting. It is worth noting that the free energy functionals used in practice are typically always nonlinear to promote phase separation in materials, and that if there are multiple species then solvers would be needed for an Allen-Cahn equation for each species. Also in the multicomponent case, the free energy functionals would need to account for interfaces between all permutations of species  $i$  and species  $j$ , when  $i \neq j$ . Interfacial terms can also be higher order terms in the local expansion of the phase field. Nonetheless, this work demonstrates applications of principles discussed in lecture to an elementary form of the Allen-Cahn equation.

## References

1. Allen, S. M. & Cahn, J. W. Ground state structures in ordered binary alloys with second neighbor interactions. *Acta Metallurgica* **20**, 423–433 (1972).
2. Allen, S. M. & Cahn, J. W. A correction to the ground state of fcc binary ordered alloys with first and second neighbor pairwise interactions. *Scripta Metallurgica* **7**, 1261–1264 (1973).
3. Biner, S. B. *Programming phase-field modeling* (Springer, 2017).
4. Provatas, N. & Elder, K. *Phase-field methods in materials science and engineering* (John Wiley & Sons, 2011).
5. Cahn, J. W. & Hilliard, J. E. Free energy of a nonuniform system. I. Interfacial free energy. *The Journal of chemical physics* **28**, 258–267 (1958).