Convection Patterns Update: PDE Solutions + ML = Profit?

Edward McDugald

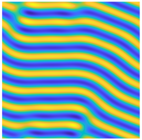
University of Arizona

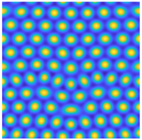
September 14, 2022

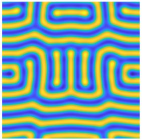
Chebfun for Easy Data Generation- Swift Hohenberg

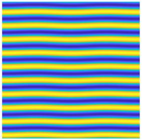
Recall, SH Reads

$$w_t = -(1+\nabla^2)^2 w + Rw - w^3$$









My attempt at numerical solution

- After reviewing the literature, I decided to try using my own "naive approach".
- I formed the discretization

$$w^{k+1} = \Delta t \left[-(2\nabla^2 + \nabla^4)w^k + (R-1)w^k - (w^k)^3 \right] + w^k$$

- I will handle tha Laplacian and Biharmonic terms using FFT

$$(2\nabla^2 + \nabla^4)w^k = \text{ifft}((2K_\Delta + K_\Delta^2)\text{fft}(w^k)).$$

My attempt at numerical solution - Matlab Code

```
function w=mySH(w, R, dt, nSteps)

[nR nC]=size(w);
if mod(nR, 2)==0
    kR=[0:nR/2-1 -nR/2:-1]*2*pi/nR;
else
    kR=[0:nR/2 -floor(nR/2):-1]*2*pi/nR;
end
Ky=repmat(kR.', 1, nC);

if mod(nC, 2)==0
    kC=[0:nC/2-1 -nC/2:-1]*2*pi/nC;
else
    kC=[0:nC/2 -floor(nC/2):-1]*2*pi/nC;
end
Kx=repmat(kC, nR, 1); % frequencies
K_Delta=Kx.^2+Ky.^2; % Fourier Laplacian
FourOp = 2*K_Delta+K_Delta.*K_Delta; % Laplacian + Biharmonic

for n=0:nSteps
    linTerm = -ifft2(FourOp.*fft2(w));
    nonLinTerm = (R-1).*w - w.^3;
    w = dt*(linTerm+nonLinTerm)+w;
end
return
```

Bonus- Ginzburg-Landau Simulations

Ginzburg-Landau Reads

$$w_t = \Delta w + w - (1 + 1.5i)w|w|^2$$