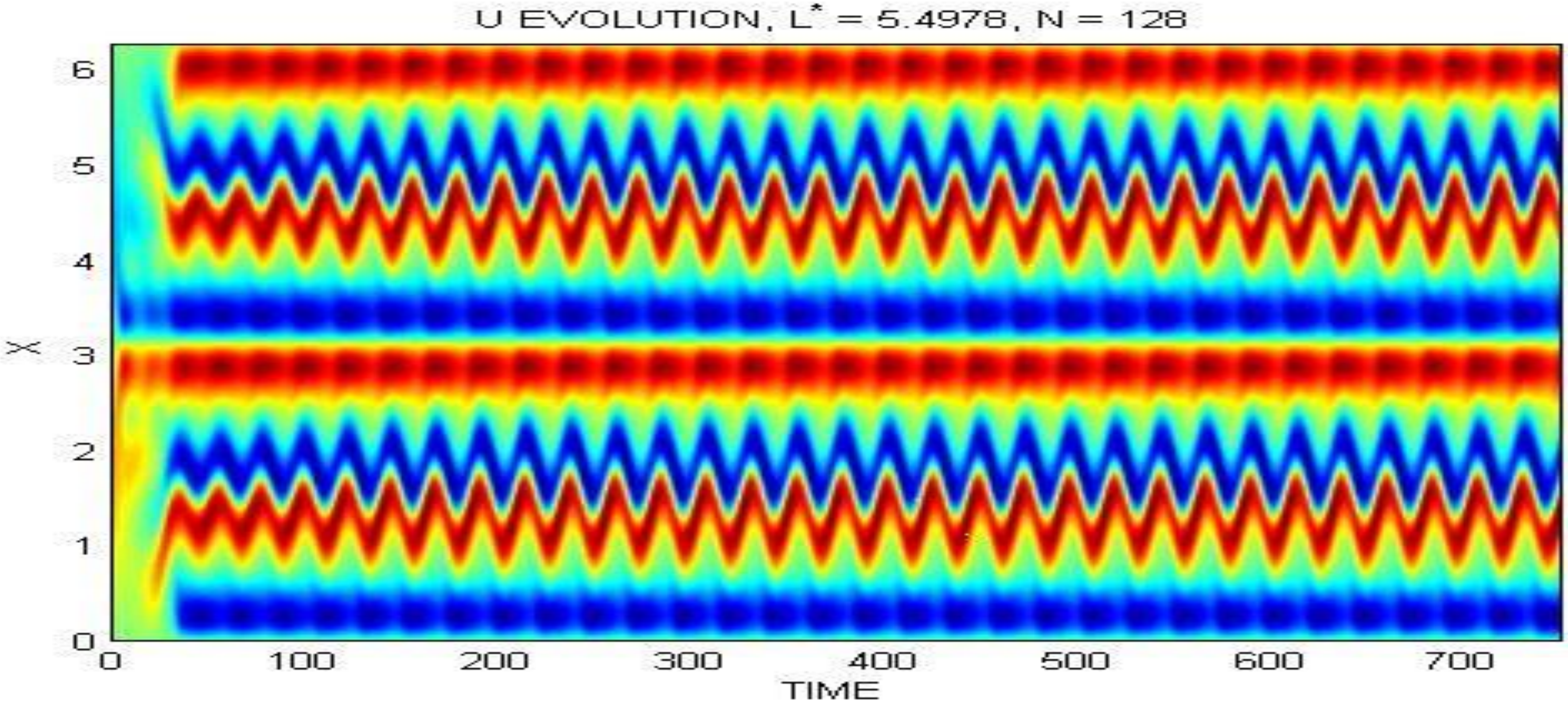
Undergraduate Research Scholar

Kyle Williams

Under the Direction of Dr. Lee Panetta



$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{\partial^2 u}{\partial x^2} + \frac{\partial^4 u}{\partial x^4} = 0$$

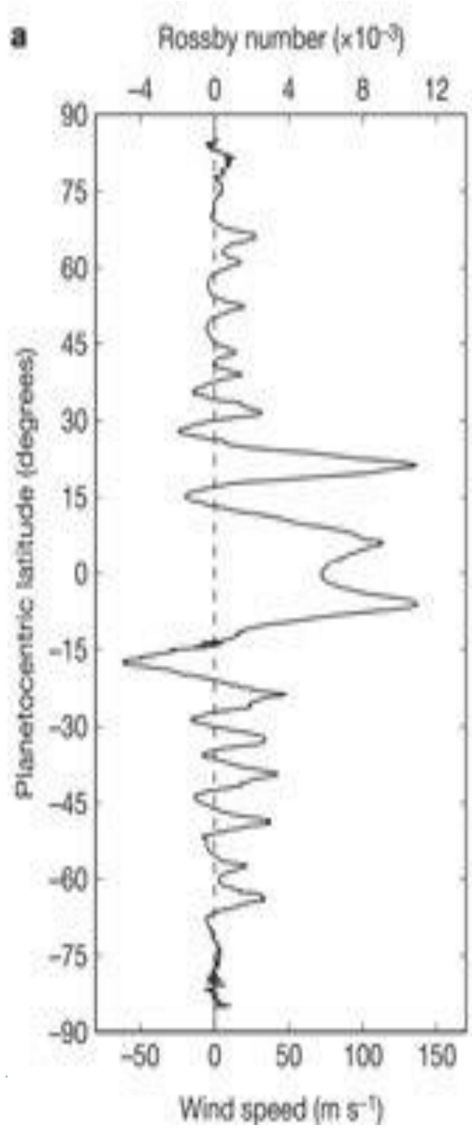
Kurimoto-Sivashinsky Equation

The Question

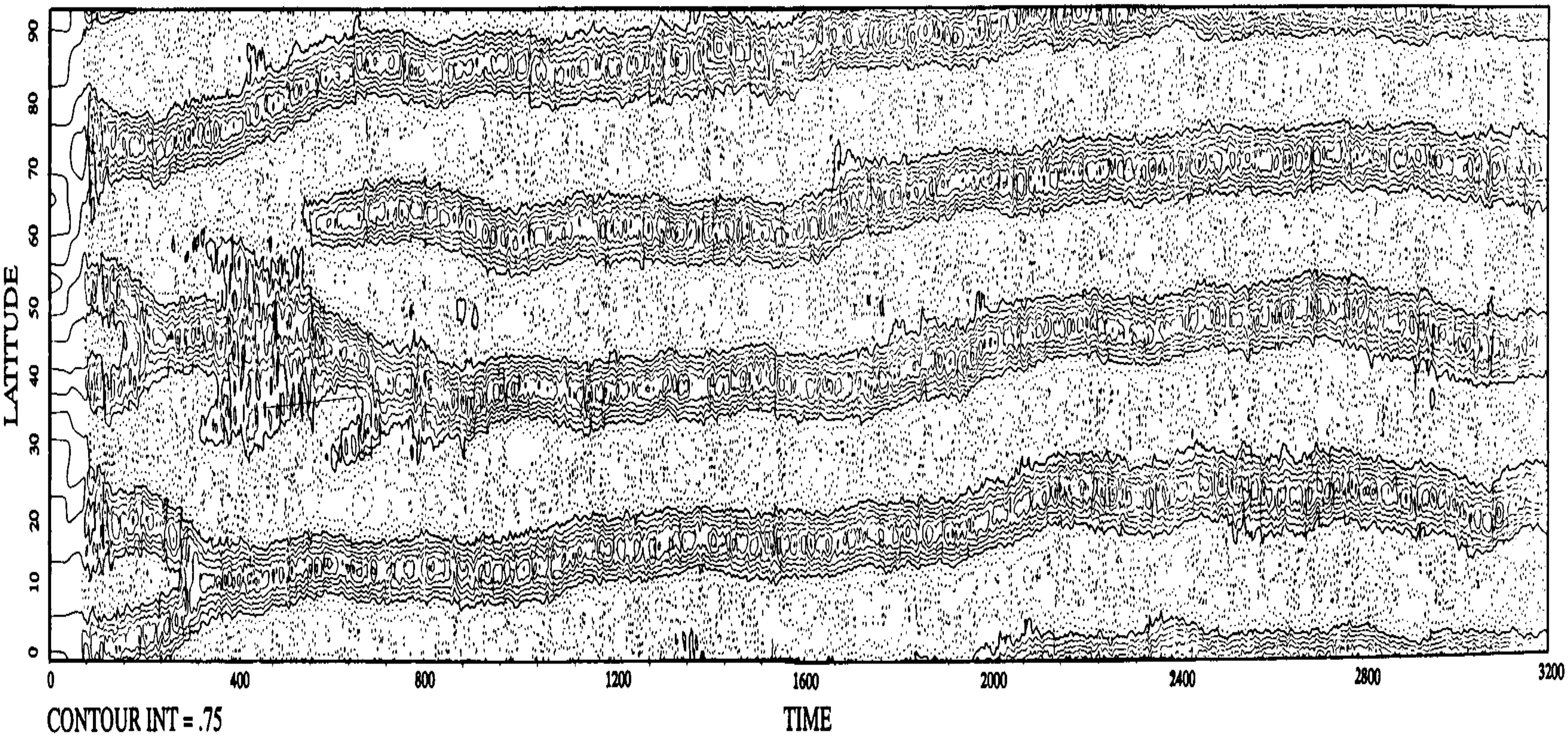
- Are numerical solutions to pattern-forming partial differential equations sensitive to time stepping methods?

- The Kurimoto-Sivashinsky Equation is a good model equation to study for this question

Has analogs in atmospheric science

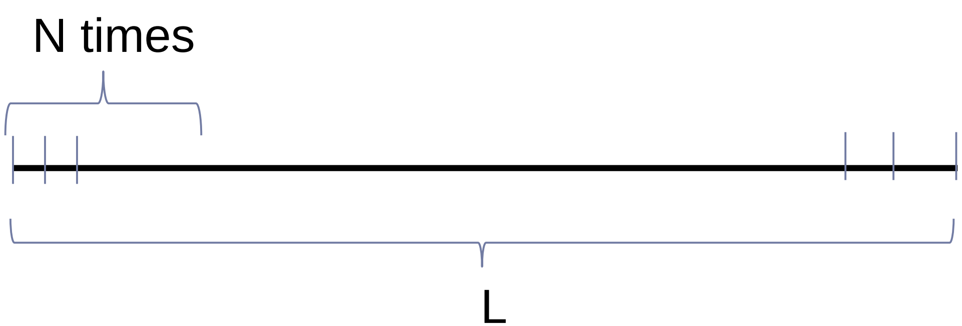


Has analogs in atmospheric science



What makes it so interesting?

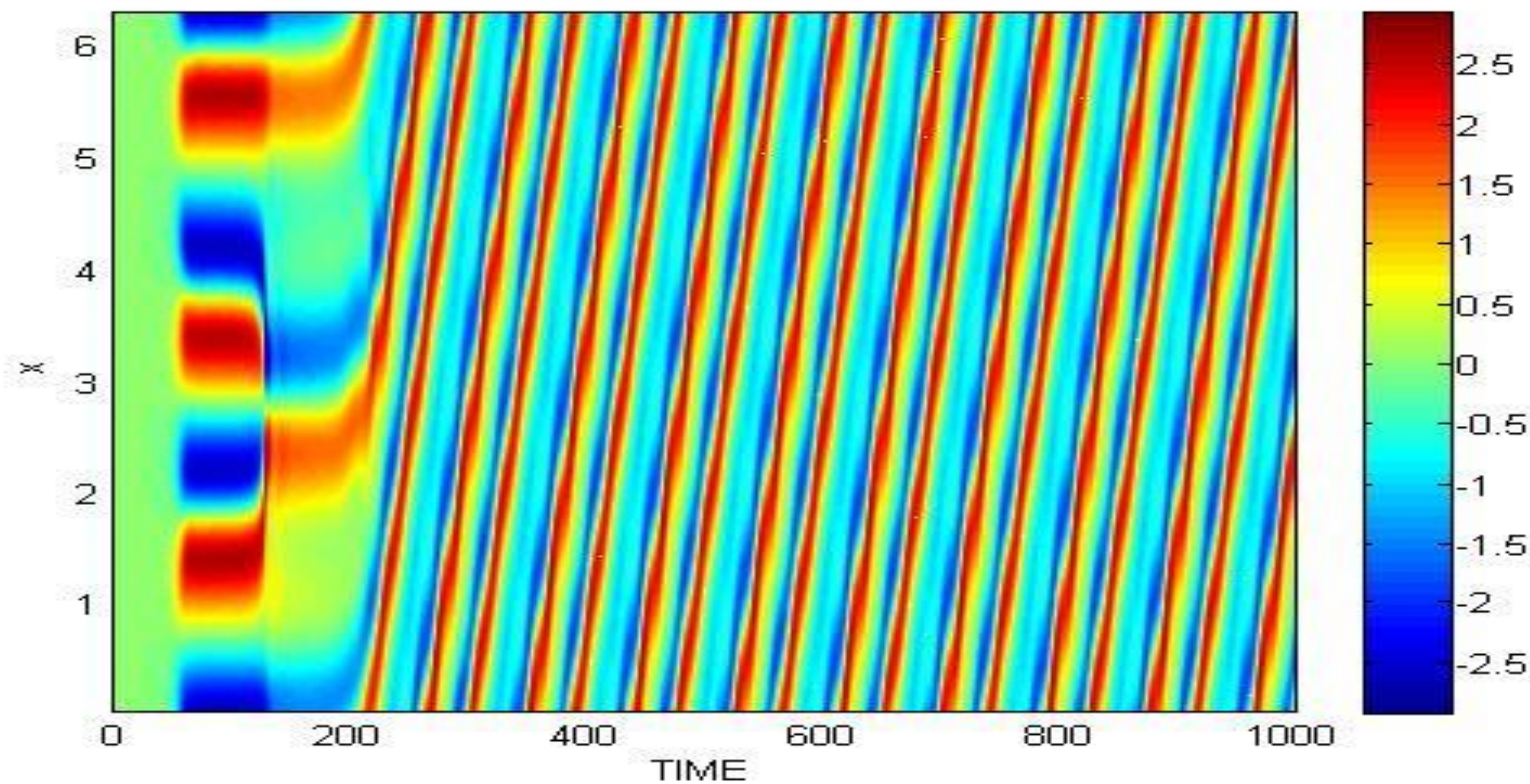
- Assume solution is periodic on interval L (common assumption in atmospheric models)
- Given initial condition $u(x,0) = u_0(x)$
- Divide L in to N parts (so N is the spatial resolution)



- For large values of $\mathrm{L}\ (> 12\pi)$, equation produces chaotic solutions
- For smaller values of L solutions have a wide array of structure
- Define $\hat{L} = \frac{L}{2\pi}$

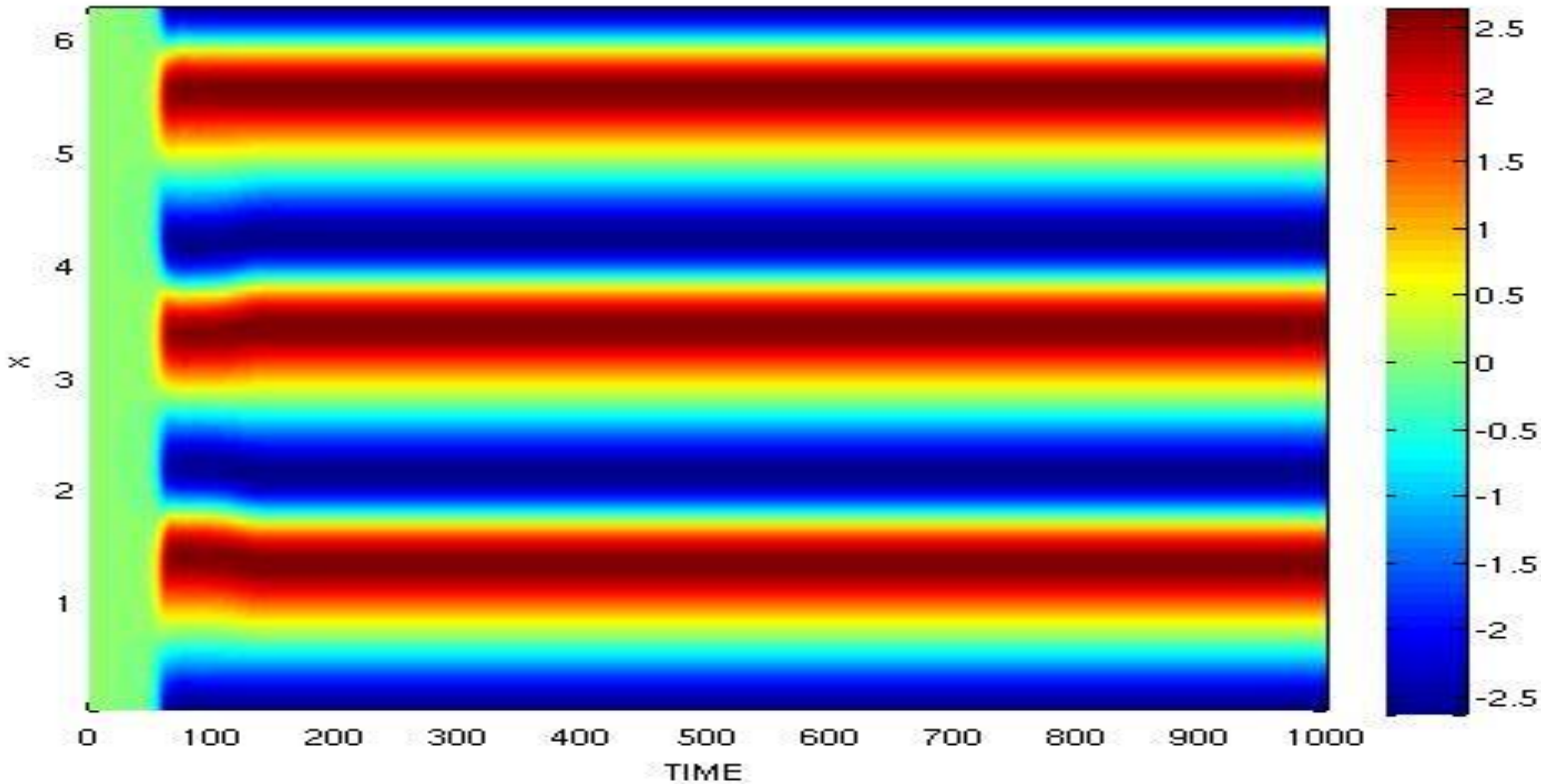
Solutions are very sensitive to $\hat{L}$

- Initial condition a randomized wave with small (10^{-5}) amplitude, $\hat{L} = 3.6398, N = 128$



Solutions are very sensitive to $\hat{L}$

- Increase $\hat{L}$ by 0.0001 with exact same initial condition



Numerical Methods

- Separate the spatial (x) and temporal (t) derivatives so it looks like

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} - \frac{\partial^2 u}{\partial x^2} - \frac{\partial^4 u}{\partial x^4}$$

- Use "pseudo-spectral" method for the spatial derivatives
- Time stepping method for time derivative

Solving the spatial derivatives

- If $u(x, t_n)$ is known, then we can use the Discrete Fourier Transform to approximate $u(x)$ as,

$$u(x, t_n) \approx \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \tilde{u}(k) e^{i\hat{k}x}$$

- Where $\hat{k} = \frac{2\pi k}{L}$
- So differentiation becomes simple multiplication

$$\frac{\partial^2 u}{\partial x^2} \approx -\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{k}^2 \tilde{u}(k) e^{i\hat{k}x} \qquad \frac{\partial^4 u}{\partial x^4} \approx \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \hat{k}^4 \tilde{u}(k) e^{i\hat{k}x}$$

Solving the spatial derivatives

- For the nonlinear term, since $u(x, t_n)$ is known and $u\frac{\partial u}{\partial x} = \frac{1}{2}\frac{\partial u^2}{\partial x}$,
- Can calculate $\frac{u^2}{2}$ and then proceed as with the linear terms.
- Let $v = \frac{u^2}{2}$
- So from the original equation,

$$\frac{\partial u}{\partial t} = -\frac{\partial v}{\partial x} - \frac{\partial^2 u}{\partial x^2} - \frac{\partial^4 u}{\partial x^4}$$

Solving the spatial derivatives

- For the nonlinear terms, since $u(x, t_n)$ is known and $u\frac{\partial u}{\partial x} = \frac{1}{2}\frac{\partial u^2}{\partial x}$,
- Can calculate $\frac{u^2}{2}$ and then proceed as with the linear terms.
- Let $v = \frac{u^2}{2}$
- We have

$$\frac{\partial \tilde{u}}{\partial t} \approx -\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} i\hat{k}\tilde{v}(k)e^{i\hat{k}x} + \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} (\hat{k}^2 - \hat{k}^4)\tilde{u}(k)e^{i\hat{k}x}$$

Spectral view of the equation

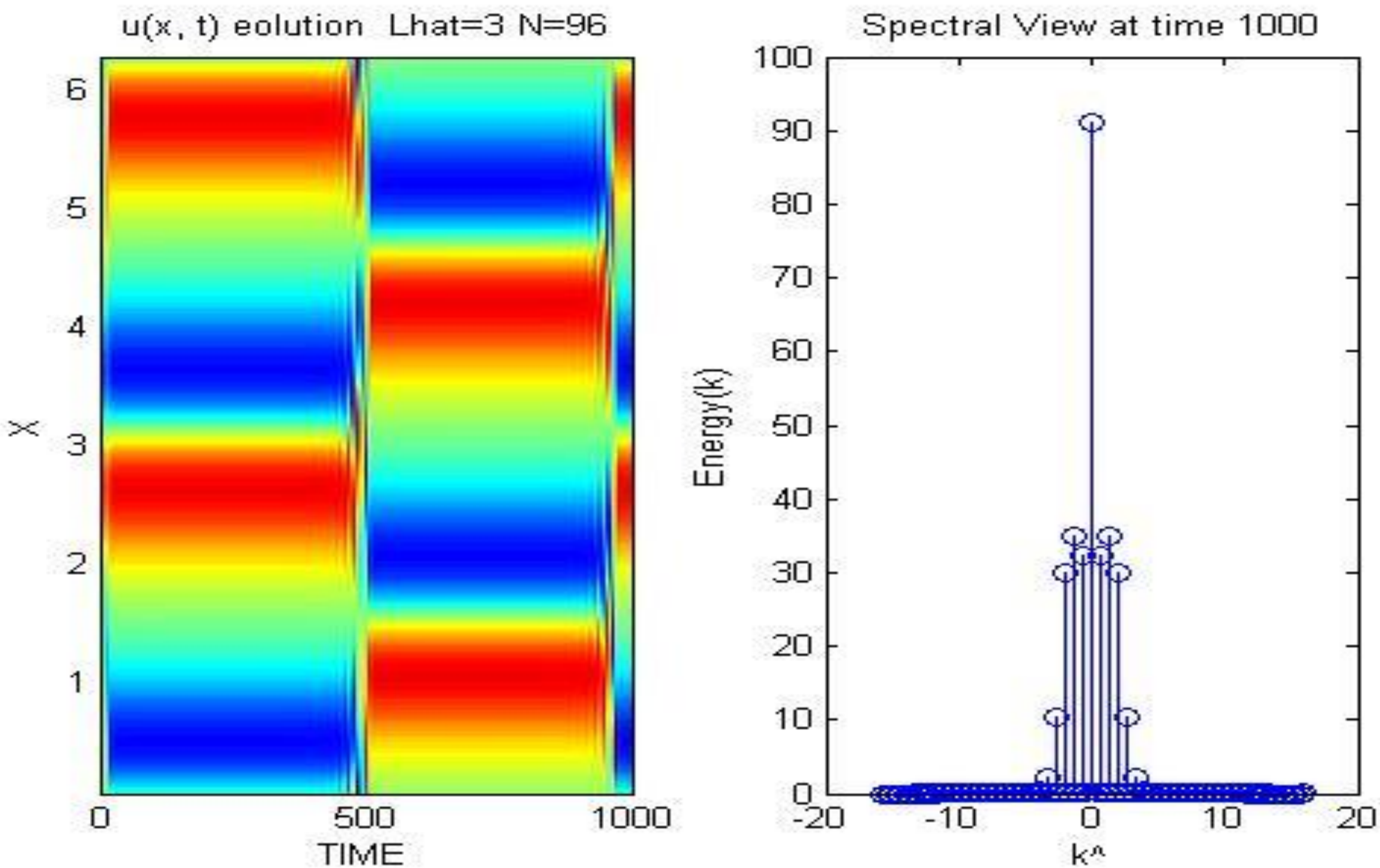
- From the spectral view of the equation,

$$\frac{\partial \tilde{u}}{\partial t} = -\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} i\hat{k}\tilde{v}(k)e^{i\hat{k}x} + \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} (\hat{k}^2 - \hat{k}^4)\tilde{u}(k)e^{i\hat{k}x}$$

- The 2nd derivative is a forcing term for low wave numbers ($|\widehat{k}| > 1$)
- The 4th derivative is a source of dissipation in the high wave numbers
- The nonlinear term transfers energy from low to high wave numbers

$$\underbrace{-u\frac{\partial u}{\partial x}}_{distributive} \quad \underbrace{-\frac{\partial^2 u}{\partial x^2}}_{forcing} \quad \underbrace{-\frac{\partial^4 u}{\partial x^4}}_{dissipative}$$

Spectral view of the equation

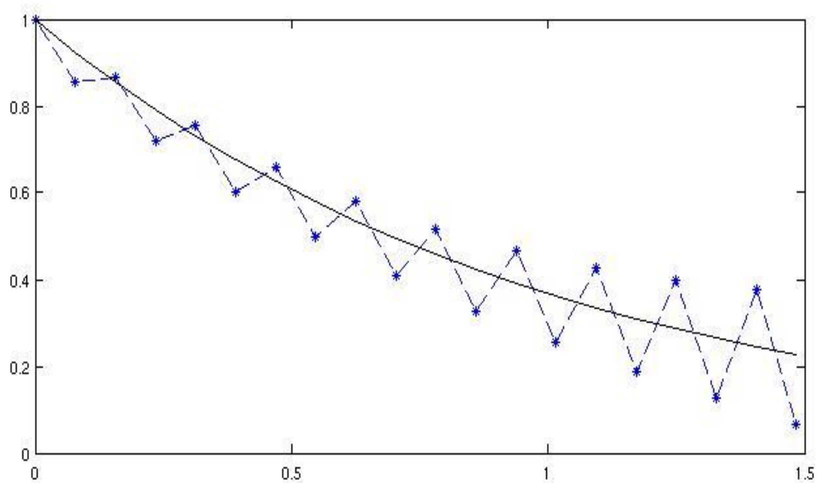


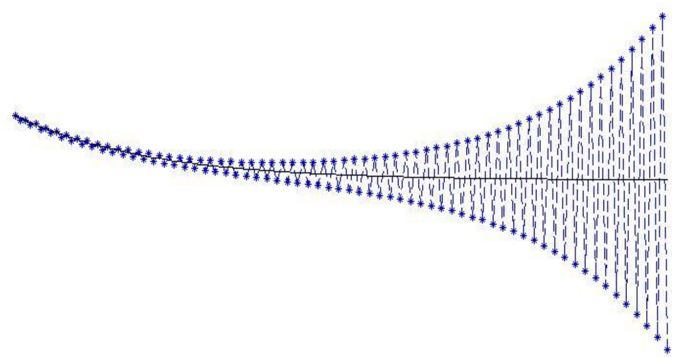
Time-Stepping methods

- Basic method:
  Leapfrog
- Two modifications
  1. Leapfrog + periodic Predictor-Corrector,
  2. Robert-Asselin-Williams (RAW) filter
- How do these two methods compare in the formation of structure in this equation?

Leapfrog time-stepping scheme

- The leapfrog scheme (centered difference) is an approximation for the time derivative
- $\frac{u(x,t_n+\Delta t)-u(x,t_n-\Delta t)}{2\Delta t} \approx \left.\frac{\partial u}{\partial t}\right|_{t_n} = f(u(x,t_n))$
- Where the right hand side is treated as a function f
- This method is unstable





Predictor Corrector method

- Stability can be improved even more by restarting every 25 steps in time.
- When starting from a single initial condition $u(t_0)$, need $u(t_0 + \Delta t)$ to use leapfrog again.
- Use Forward Euler method* to calculate $u\left(t_0 + \frac{\Delta t}{2}\right)$
- Use leapfrog on $u(t_0)$ and $u\left(t_0 + \frac{\Delta t}{2}\right)$ to calculate $u(t_0 + \Delta t)$ then continue as before
- *Forward Euler: $\left.\frac{\partial u}{\partial t}\right|_{t_n} \approx \frac{u(t_n + \Delta t) - u(t_n)}{\Delta t}$

Robert-Asselin-Williams Filter

- A separate improvement on the basic leapfrog scheme is the RAW filter

$$\frac{u(x,t_n+\Delta t)-\overline{\overline{u(x,t_n-\Delta t)}}}{2\Delta t}=f\left(\overline{u(x,t_n)}\right)$$

- Where,

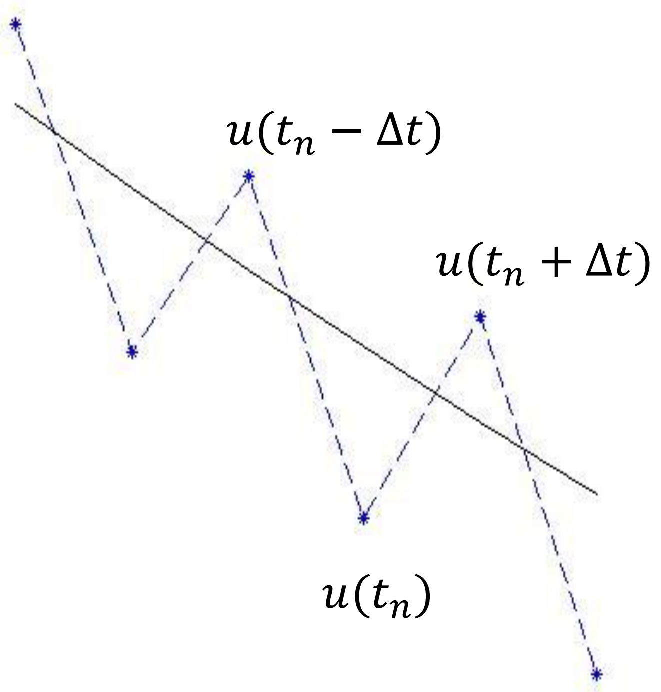
$$\overline{u(x,t_n)}=u(x,t_n)+\frac{\nu(1-\alpha)}{2}\left[\overline{\overline{u(x,t_n-2\Delta t)}}-2\overline{u(x,t_n-\Delta t)}+u(x,t_n)\right]$$

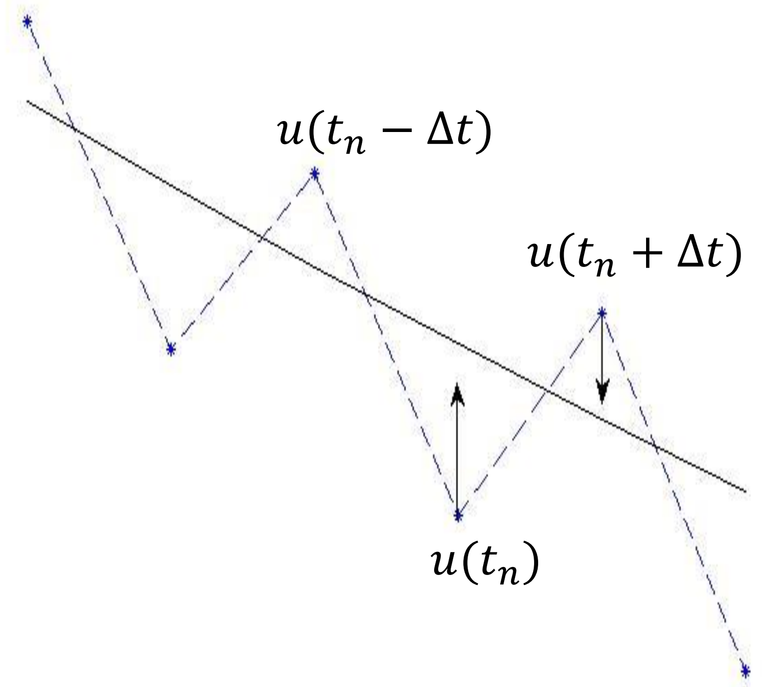
- And,

$$\overline{\overline{u(x,t_n-\Delta t)}}=\overline{u(x,t_n-\Delta t)}-\frac{\nu\alpha}{2}\left[\overline{\overline{u(x,t_n-2\Delta t)}}-2\overline{u(x,t_n-\Delta t)}+u(x,t_n)\right]$$

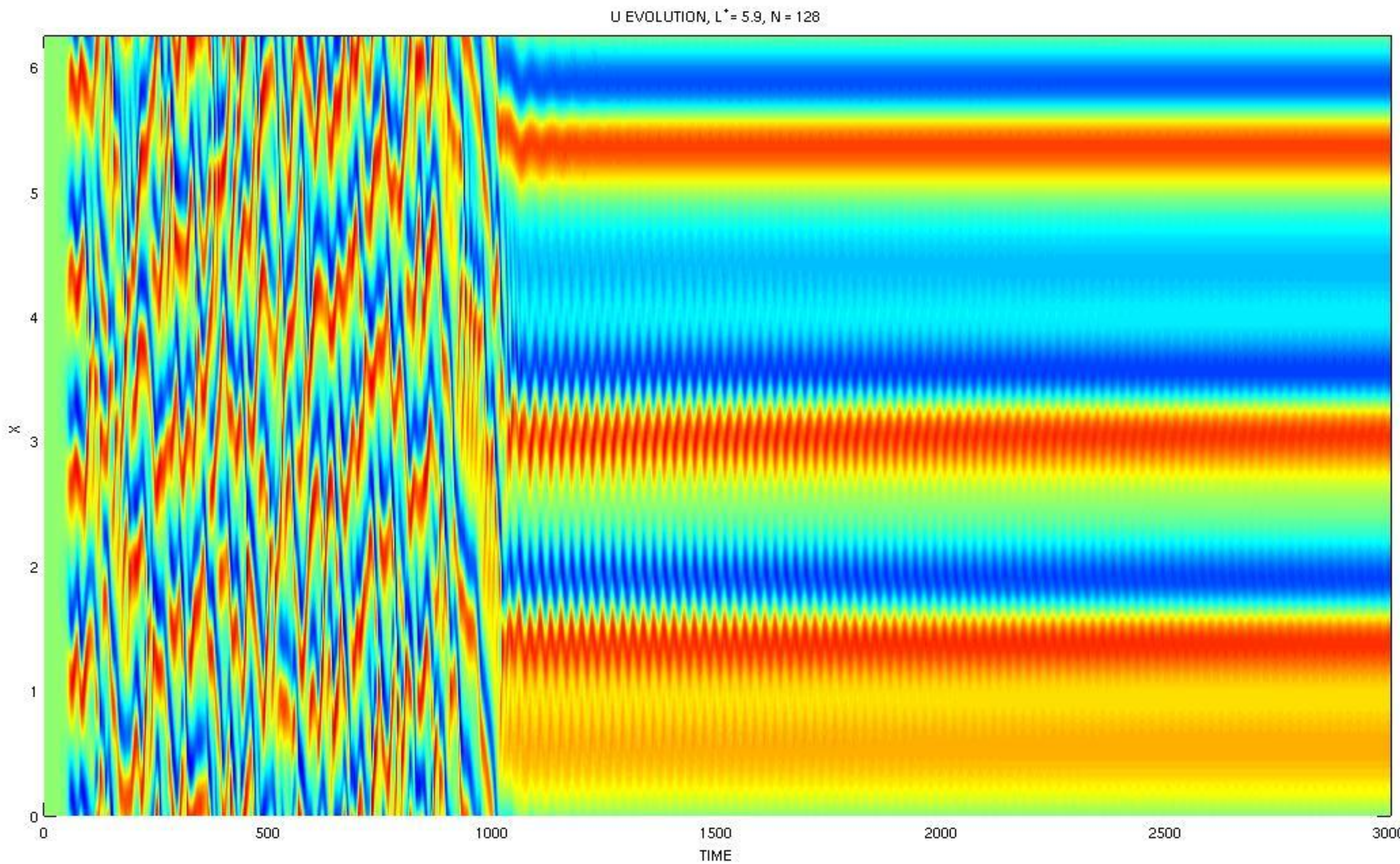
Robert-Asselin-Williams Filter

- What this is saying is we compute the next time step
- Then push $u(t_n)$ and $u(t_n + \Delta t)$ towards the

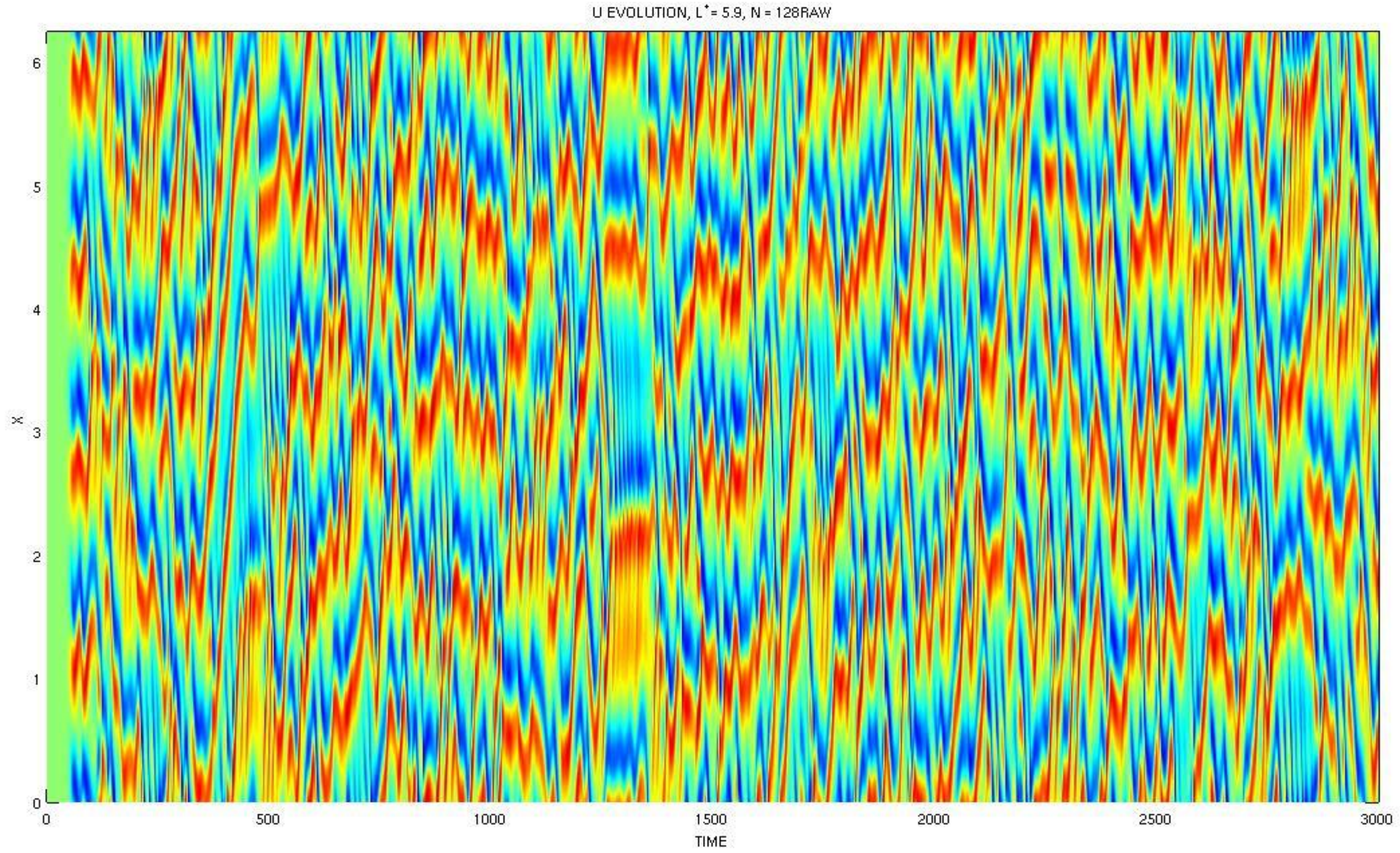




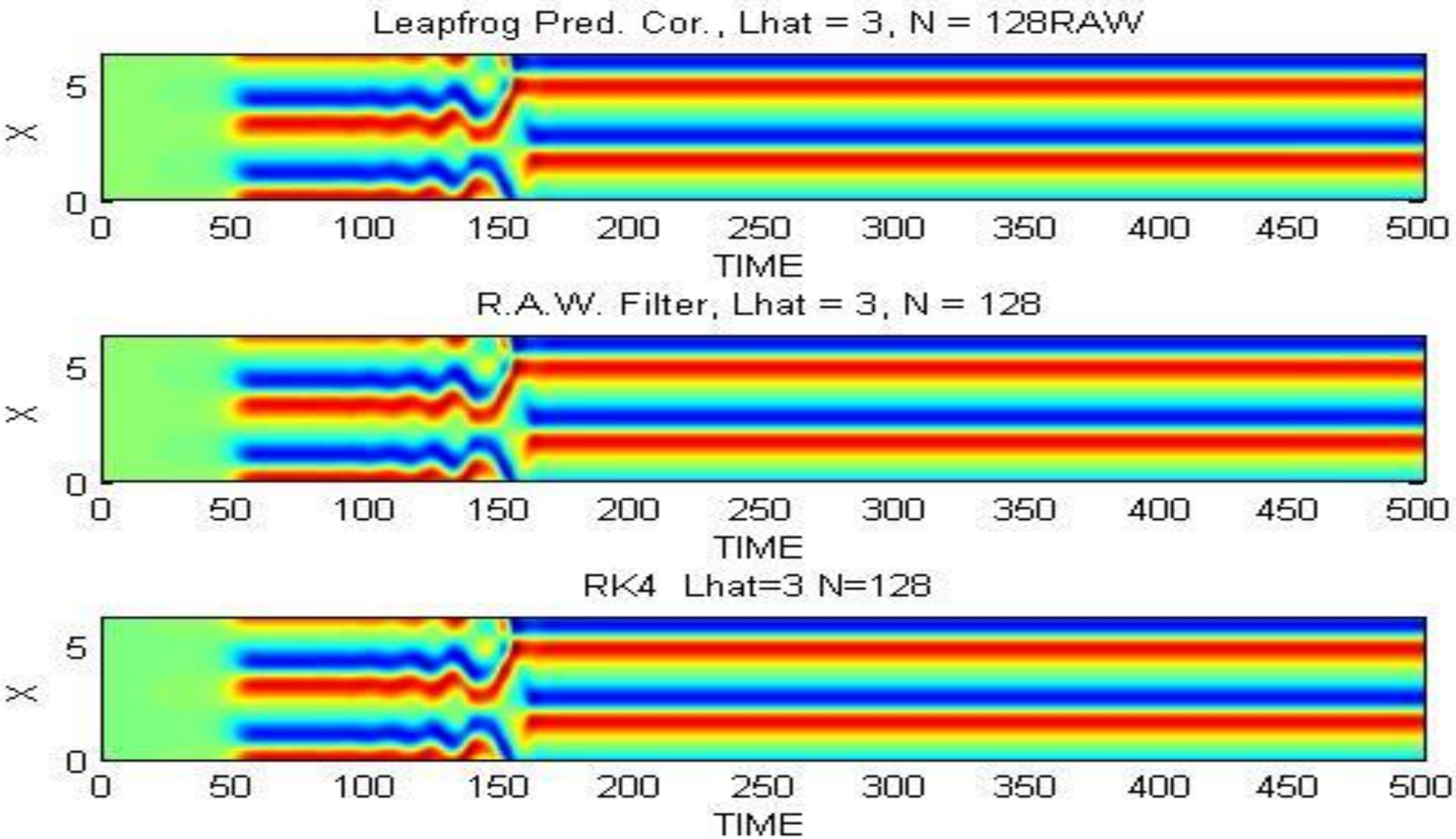
What is the problem?



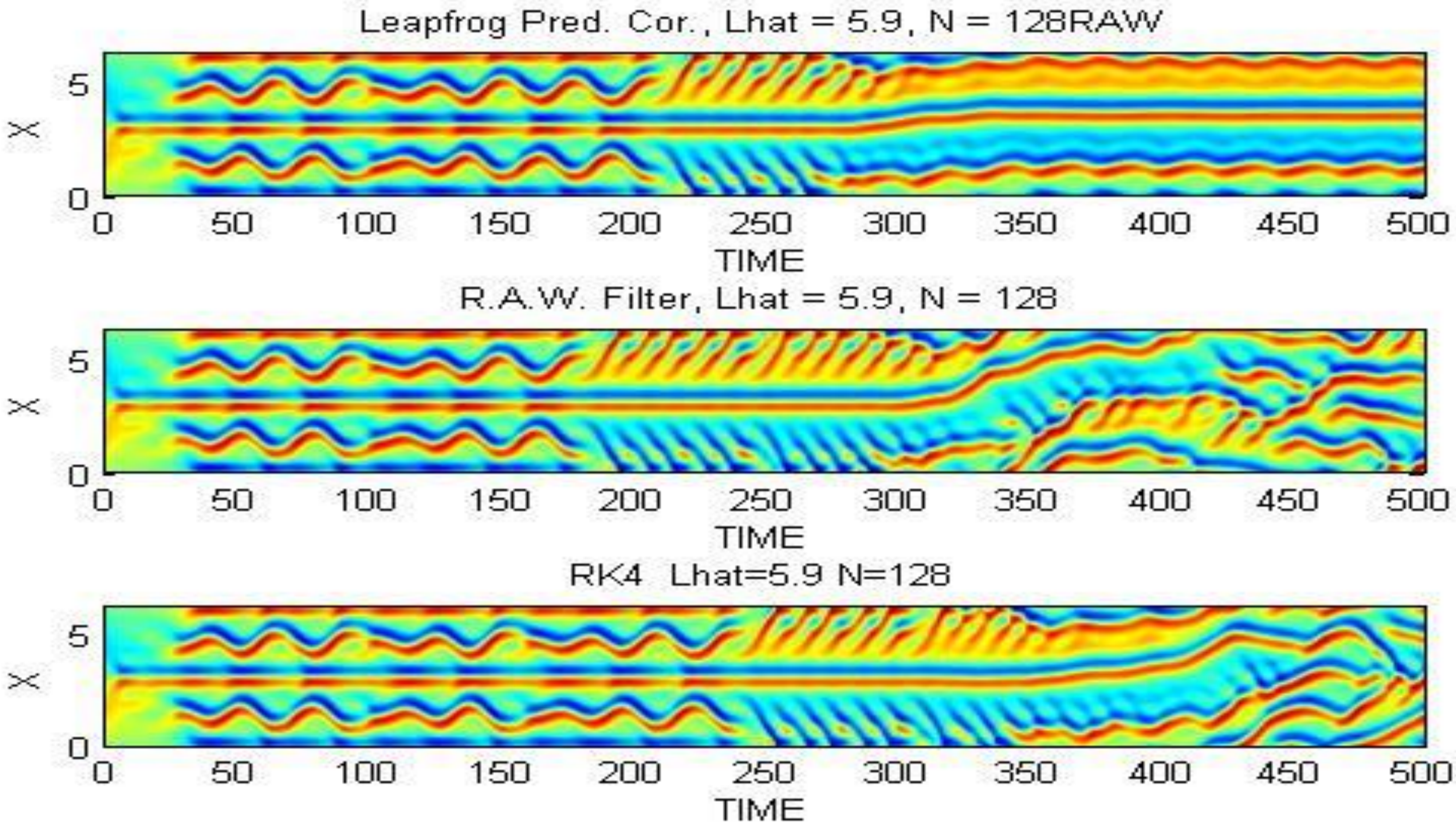
What is the problem?



Compared to more accurate method



Compared to more accurate method



Conclusion: Time-stepping method matters

- 4^{th} order Runge-Kutta method takes time and memory
- Can also be difficult to implement in existing code

- In general RAW gives a better idea of the behavior of the solution than the Predictor Corrector method
- It is also very simple to update existing code

- Williams has produced a more general filter to give up to 7^{th} order accuracy

- When looking at the development of structure time-stepping matters!