Stability Concerns in the Integrating Factor Method

Jeff Haack

May 11, 2007

1 Introduction

The pseudospectral integrating factor method is a numerical method used for solving equations of the form

$$u_t + \mathcal{N}(u) + \mathcal{L}(u) = 0$$

Where $\mathcal{N}$ is a nonlinear operator and $\mathcal{L}$ is a linear operator, with periodic boundary conditions.

The problem with 'usual' methods for solving such equations is that the linear part occurs on a much shorter timescal than the nonlinear part, requiring one to spend a great amount of computational effort to observe long-time phenomena caused by the nonlinear term. The method seeks to remove this difficulty by introducing an integrating factor to remove the stiff linear part from the equation. In Fourier space this results in

$$\hat{u}_t + \mathcal{F}\,\mathcal{N}(u) + i\omega(k)\hat{u} = 0$$

An integrating factor is made out of the linear part by defining

$$\hat{U}(k,t) = e^{i\omega(k)(t-t_n)}\hat{u}$$

Thus the equation to be evolved is

$$\hat{U}_t = -e^{i\omega(k)(t-t_n)}\mathcal{F}\,\mathcal{N}(e^{-i\omega(k)(t-t_n)}\hat{U})$$

This equation can now be evolved in time using Runge-Kutta or another appropriate method, while dealing with any spatial derivatives using fast fourier transforms. Computation time is saved by treating t_n in the above equation to be the most recent time during the time evolution, thus the integrating factor $e^{\omega(k)(t-t_n)}$ is the same at each timestep.

Now that the linear part is effectively removed from the evolution equation, the stability is expected to hinge on the less stiff nonlinear operator, allowing larger timesteps and thus quicker computation.

In the following numerical examples, a fourth order Runge-Kutta scheme is used to numerically solve the Korteweg-deVries equation (1) in the periodic domain $[-\frac{L}{2}\frac{L}{2}]$.

$$u_t + \frac{1}{2}(u^2)_x + u_{xxx} = 0 \tag{1}$$

For KdV, $\omega(k) = -k^3$ and the evolution equation takes the form

$$\hat{U}_t(k,t) = -\frac{ik}{2}e^{-ik^3(t-t_n)}\mathcal{F}(\mathcal{F}^{-1}(e^{ik^3(t-t_n)}\hat{U}))^2$$

KdV admits solitary wave solutions of the form

$$u(x,t) = 3csech^2(\frac{\sqrt{c}}{2}(x-ct))$$

Fixed parameters for the following simulations are $L = 60$ and $c = 2.25$.

After thorough testing of the method on the KdV equation, we find that the method does not eliminate the stiffness from the problem. However, it does change the nature of the instability that we see, and ensures that the instability barrier is much softer in a sense than the timestep restriction in a direct psudospectral methad.

2 Initial results

First, the error scalings with respect to spatial and time discretizations are verified.

Taking fixed values of Δt and varying values of Δx results in Figure 1. The envelope of this error graph looks like $\frac{1}{\Delta x}$ which is to be expected, as this should be spectrally accurate in space owing to how we approximate our spatial derivatives. The slight waviness of the error graph is due to the changing grid spacing for different Δx.

Taking a fixed Δx and varying Δt results in Figure 2. Aside from one troubling instability, there is a smooth error trend until a critical timestep, where the errors become less regular. This critical timestep is assumed to be the onset of instability of the method. The extra instability, occuring before the critical timestep, will need to be accounted for. The label 'A' shows this instability as it appears on both graphs.

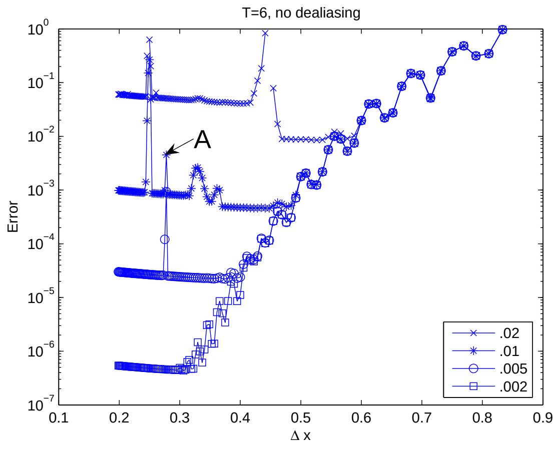

Figure 1: T = 6, 4 choices of Δt, Δx varying

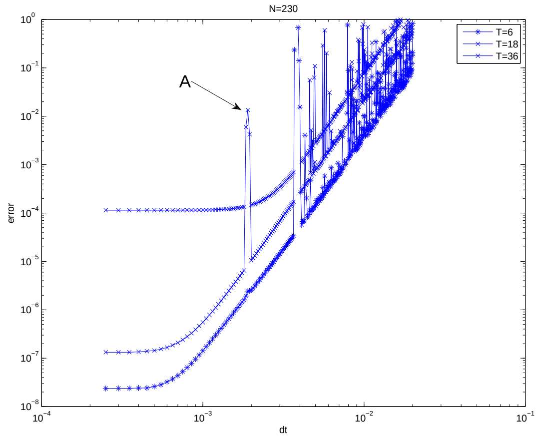

Figure 2: $\Delta x = .26$ (N=230) for 3 final times

There are 2 possiblilites for what may be causing these instabilities

1. One (easy to fix) possibility is aliasing instabilities, that sometimes come up in spectral evaluation of nonlinear terms. Adding dealiasing will increase the amount of computational time spent on this method by $\approx 50\%$

2. Another possibility is that some sort of discrete resonance is causing some modes to gain too much energy. This comes from a coupling between wavenumbers in the phase factor term on the right hand side, which for this problem becomes

$$e^{i\Delta t(p^3+q^3-k^3)}$$

This can also be seen by noting that the first wavelength the method cannot resolve in the time integrations is $\frac{\pi}{\Delta t}$. Thus if $p^3 + q^3 - k^3 = \frac{2\pi}{\Delta t}$ resonance instabilities should be expected in the expression above.

3 Attempts to address the instabilities

3.1 Dealiasing

Atfer dealiasing in included in the method, the error graphs take the form

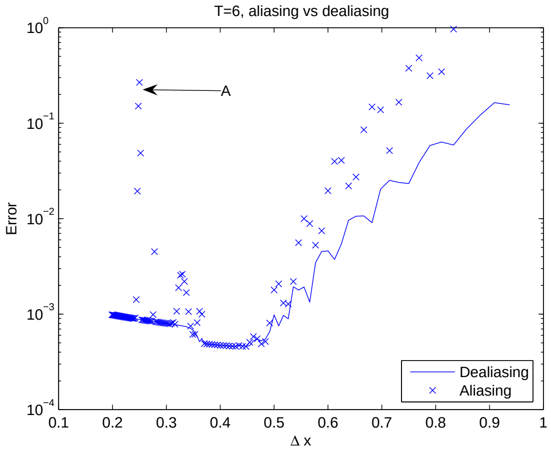

Figure 3: $\Delta t = .005,\ T = 6$

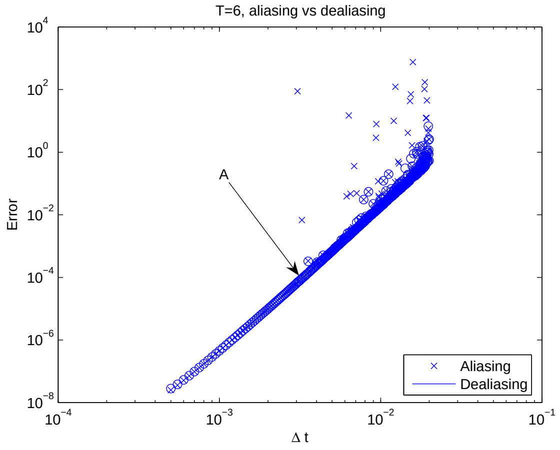

Figure 4: $\Delta x = .26,\ T = 6$

Things are much cleaner now. In the Δx fixed graph, dealiasing has removed the instability before the critical timestep, and the Δt fixed graph is much more regular now as well.

If we look at long times we confirm that there is instability past the critical timestep. However, these instabilities take a very long time to develop (see Figure 5)

Now that the spurious instabilities have been addressed, a power law can be found for the critical timestep at each spatial discretization. For example, Figure 6 shows the first spike for N = 256, which occurs at $\Delta t = .0031$. The critical timesteps found are plotted against Δx in Figure 7. Alarmingly, this graph shows a third order scaling for the stability limit. This fundamentally challenges the motivation for the entire method, as the unmodified problem had third order stability scaling corresponding to the third order derivative.

3.2 Artificial Resonance

The phase factor in the right hand side of the evolution equation is $\Omega = \omega(k) - \omega(q) - \omega(p)$, where $p + q = k$. As mentioned above, resonance is expected when $\Omega = \frac{2\pi}{\Delta t}$. Because $\omega(k) = k^3$ for the KdV equation, $\Delta t_{res} = O(\Delta x^3)$.

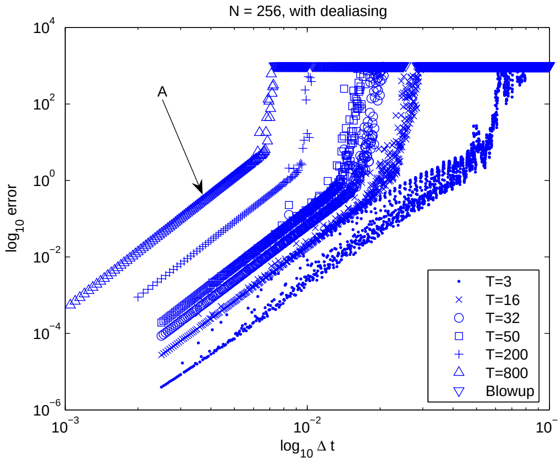

Figure 5: Errors at T=3, T=16, T=32, T=50, T=200, T=800

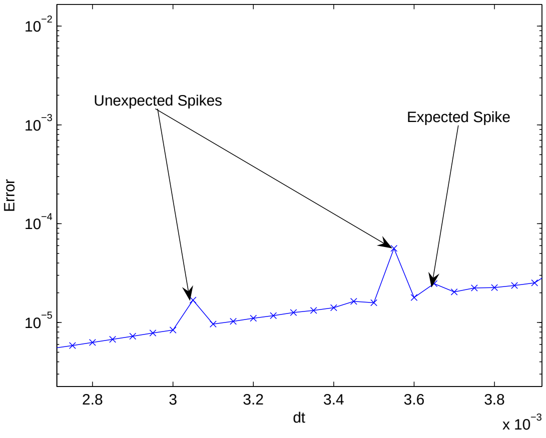

Figure 6: First spike, N = 256 (with dealiasing)

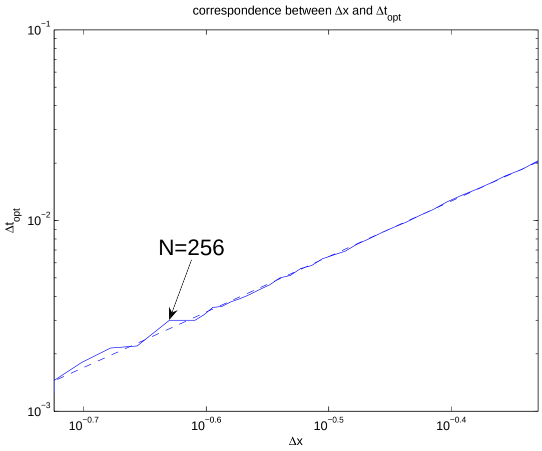

Figure 7: Third order scaling of instability: timestep vs spatial resolution

A quick calculation with $N = 256$ ($\Delta x = .234$) suggests a first instability of $\approx .00365$. This does not however quite line up with the first observed spike in the graph which is at $\Delta t = .00315$ (Figure 6) (Note: all the following graphs are with N=256)

One guess as to why we this does not line up is that since RK4 uses different timesteps as it evolves the equation, but it is not exactly clear how to analyze how the method affects this scheme. We belive that this resonance is the mechanism that is causing the instabilities.

3.3 Attemps to get around the stability limit

A few modifications were made to the method to get around this third order scaling, but all were unsuccessful

- One might expect a random timestepping scheme to scramble the resonant terms enough to fix the stability. However, this failed to be the case in practice.
- A coarse-graining approach, which uses a large, unstable timestep and a small, stable timestep might also serve to damp out these unstable resonances. However, using this method allots most of one's time to the small, stable timestep which does nothing to alleviate the computational cost of the problem.
- A Symplectic integrator would be expected to conserve some quantities of the equation. However, aside from the fact that implicit methods require more computational time to evolve, a simple fixed point method for finding the implicit term shows a strong dependence on the timestep in convergence. One can find a larger stable timestep than the explicit method by using this method, but the computational time required is actually larger than the stiff explicit method. One could perhaps make a newton solver, but creating a newton solver for the complicated RHS can be difficult and costly.

4 Summary and Conclusions

While the stiffnes issue has not been resolved by the method, we can see from Figure 5 that although the method becomes unstable for a large range of timesteps, the instabilites often take a lot of time to develop. Comparing this with a similar pseudospectral method without an integrating factor (Figure 1), we see that the direct method shows its stability limit right away, while the integrating factor method takes some time to blow up.

Final Time	Direct Method	Integrating Factor Method
1	.0012	.5
3	.0012	.0795
10	.0012	.0392
50	.0012	.0177
200	.0012	.0104
800	.0012	.0072

Table 1: Largest stable timestep for various final times (N = 256)

Thus we can say that although the stiffness problem has not truly been resolved, the integrating factor method is more suitable for solving such wave-like systems provided that we choose a timestep suitable for the amount of evolved time.

Another option is to look at stiff ODE solvers, such as Radau solvers, and bypass this integrating factor business altogether.