Linearly Stabilized Schemes for the Time Integration of Stiff Nonlinear PDEs

Kevin Chow

December 9, 2016

Table of contents

Introduction

- Goal: time stepping for stiff nonlinear PDEs.
  - Stability
  - Accuracy
  - Simplicity
  - Efficiency

Example

Consider the heat equation,

$$u_t = u_{xx}, \quad x \in \Omega, \quad t > 0.$$

Discretize in space:

$$U' = LU, \quad U \in \mathbb{R}^N, \quad t > 0.$$

Explicit: $U^{n+1} = G(U^n, U^{n-1}, \ldots, LU^n, LU^{n-1}, \ldots)$, but $\Delta t \leq Ch^2$.
Implicit: $AU^{n+1} = b$; unconditionally stable, but must solve a linear system.

Example

Now compare with

$$u_t = \frac{u_{xx}}{1+u_x^2} - \frac{1}{u}, \quad x \in \Omega, \quad t > 0.$$

and

$$U' = F(U), \quad U \in \mathbb{R}^N, \quad t > 0.$$

Explicit: $U^{n+1} = G(U^n, U^{n-1}, \dots, F(U^n), F(U^{n-1}), \dots)$, but $\Delta t \le Ch^2$.
Implicit: $AU^{n+1} = b(U^{n+1})$; unconditionally stable, but must solve a nonlinear system because nonlinearity is in the stiff term.

Example

Comparing side-by-side:

$$u_t = u_{xx}, \quad x \in \Omega, t > 0,$$

Explicit: $\Delta t \leq Ch^2$
Implicit: unconditionally stable; solution to linear system

$$u_t = \frac{u_{xx}}{1 + u_x^2} - \frac{1}{u}, \quad x \in \Omega, t > 0,$$

Explicit: $\Delta t \leq Ch^2$
Implicit: unconditionally stable; solution to nonlinear system

Summary: What We Like

Explicit: simple; handles nonlinear terms with no added difficulty.
Implicit: large time steps

Example

Modify the equation,

$$u_t = \frac{u_{xx}}{1+u_x^2} - \frac{1}{u} - u_{xx} + u_{xx}, \quad x \in \Omega, \quad t > 0,$$

and discretize in space,

$$U' = F(U) - LU + LU, \quad U \in \mathbb{R}^N, \quad t > 0.$$

Use implicit-explicit time stepping, e.g.

$$\frac{U^{n+1} - U^n}{\Delta t} = F(U^n) - LU^n + LU^{n+1}.$$

Linear Stability

More generally, from $U' = F(U)$, we can modify as

$$U' = \underbrace{F(U) - pLU}_{(\star)} + pLU, \quad p > 0,$$

and apply a time stepping scheme that treats $(\star)$ explicitly.

Is this unconditionally stable?

Scalar test equation

Standard case:

$$U' = F(U)$$

Linearize → Diagonalize → Test equation:

$$u' = \lambda u$$

With linear modification:

$$U' = F(U) - pLU + pLU$$

Linearize → Diagonalize → Test equation:

$$\begin{aligned} u' &= \lambda u - p\lambda u + p\lambda u \\ &= (1-p)\lambda u + p\lambda u \end{aligned}$$

Apply time stepping method:

$$u^{n+1} = \xi(\lambda \Delta t) u^n.$$

Unconditional stability:

$$|\xi(\lambda \Delta t)| \leq 1 \quad \text{for all} \quad \lambda \Delta t < 0.$$

Implicit-explicit Euler

Applied to the test equation, $u' = (1-p)\lambda u + p\lambda u$, yields

$$\frac{u^{n+1} - u^n}{\Delta t} = (1-p)\lambda u^n + p\lambda u^{n+1}.$$

The amplification factor is

$$\xi_1(\lambda\Delta t) = \frac{1 + (1-p)\lambda\Delta t}{1 - p\lambda\Delta t}.$$

Impose unconditional stability:

$$|\xi_1(\lambda\Delta t)| \leq 1 \text{ for all } \lambda\Delta t < 0 \iff p \geq 1/2.$$

Explicit-implicit-null (EIN)

Duchemin and Eggers (2014) use Richardson extrapolation to get second order. The amplification factor is

$$\xi_{EIN}(\lambda\Delta t) = 2\xi_1^2(\lambda\Delta t/2) - \xi_1(\lambda\Delta t).$$

and

$$|\xi_{EIN}(\lambda\Delta t)| \leq 1 \text{ for all } \lambda\Delta t < 0 \iff p \geq 2/3.$$

Implicit-explicit multistep methods

An alternative for second and higher order methods: IMEX multistep methods.

Table : Parameter restriction for select IMEX methods.

Order	Method	$p \in$
1	IMEX-Euler	$[1/2, \infty)$
2	SBDF2	$[3/4, \infty)$
	CNAB	$[1, \infty)$
	mCNAB	$[8/9, \infty)$
	CNLF	$[1/2, \infty)$
3	SBDF3	$[7/8, 2]$
4	SBDF4	$[11/12, 5/4]$

Understanding the methods: 3 key properties

Do the methods work as advertised?
Examine this with two test problems,

$$u_t = \frac{u_{xx}}{1+u_x^2} - \frac{1}{u},$$

and

$$u_t = \Delta(u^5).$$

Test Problem 1

First test problem:

$$u_t = \frac{u_{xx}}{1+u_x^2} - \frac{1}{u}, \quad 0 < x < 10, \quad t > 0,$$

with initial condition

$$u(x,0) = 1 + 0.10 \sin\left(\frac{\pi}{5}x\right),$$

and boundary conditions $u(0,t) = 1 = u(10,t)$.
Stabilized as

$$u_t = \frac{u_{xx}}{1+u_x^2} - \frac{1}{u} - pu_{xx} + pu_{xx}.$$

Numerical convergence test

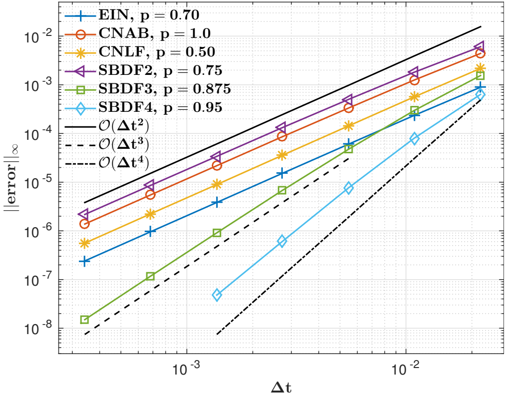

Figure : Numerical convergence of linearly stabilized schemes at time $T = 0.35$. Compared against a reference solution generated using an explicit 3rd order RK with $\Delta t = 1.46 \times 10^{-5}$. Stabilized by adding and subtracting pu_{xx}.

Selecting p for more general problems

How did we choose p? Consider

$$u' = \lambda u - p\lambda u + p\lambda u$$

and

$$U' = F(U) - pLU + pLU.$$

With the test equation, we derived a restriction on p. More generally, the restriction applies to $p\lambda_L/\lambda_F$. For test problem 1 with centred differences, we find

$$\frac{p\lambda_L}{\lambda_F} \approx p(1 + (D_1 \bar{u}_j^n)^2).$$

Rejecting methods with bounded p-parameter restrictions

The selection of p for SBDF3 is dictated by

$$\max_{1\leq j\leq N} \frac{7}{8}\frac{1}{1+(D_1\bar{u}_j^n)^2} \leq p \leq \min_{1\leq j\leq N} \frac{2}{1+(D_1\bar{u}_j^n)^2}.$$

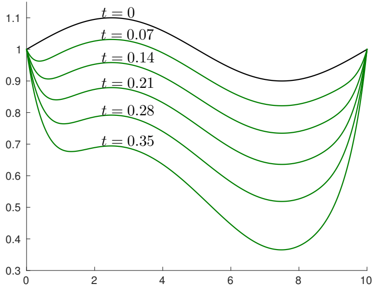

Figure : Numerical solution to test problem 1 .

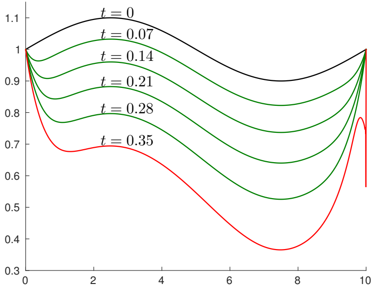

Figure : Development of instabilities using SBDF3, $p = 1.625$.

Test Problem 2

Second test problem:

$$u_t = \Delta(u^5), \quad (x, y) \in [0, 1]^2, \quad t > 0,$$

with initial and boundary conditions set such that the exact solution is

$$u(x, y, t) = \left(\frac{4}{5}(2t + x + y)\right)^{1/4}.$$

Stabilize with $p\Delta u$; $p\lambda_L/\lambda_F \approx p/(8(1 + t))$.

Numerical convergence test

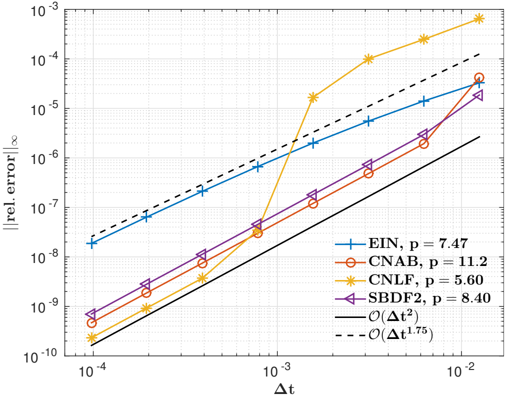

Figure : Numerical convergence of linearly stabilized schemes at time $T = 0.40$. Compared against a reference solution generated using an explicit 3rd order RK with $\Delta t = 6.25 \times 10^{-6}$. Stabilized using $p\Delta u$; $p\lambda_L/\lambda_F \approx p/(8(1+t))$.

Error coefficient and dependence on p

Discretizing $u' = (1-p)\lambda u + p\lambda u$, we observe that the discretization error grows with p.

How does the error behave as we increase p?

Examine the coefficient of the leading order error term.

Table : Coefficient of leading order error term as applied to the test equation.

Method	Coefficient
EIN	$\frac{1}{2}(p - p^2)$
SBDF2	$\frac{2}{3}p - \frac{5}{18}$
CNAB	$\frac{1}{2}p - \frac{1}{4}$
CNLF	$p - \frac{1}{6}$

Large values of p needed to stabilize

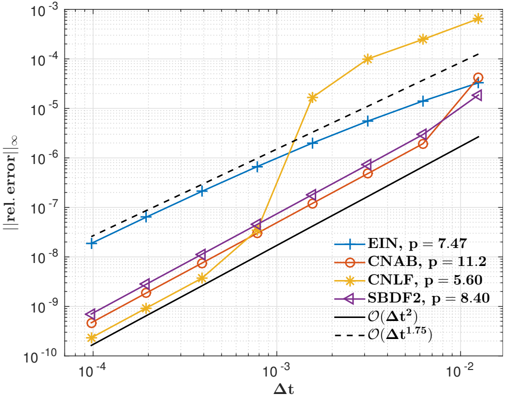

Figure : Numerical convergence of linearly stabilized schemes at time $T = 0.40$. Compared against a reference solution generated using an explicit 3rd order RK with $\Delta t = 6.25 \times 10^{-6}$. Stabilized using $p\Delta u$; $p\lambda_L/\lambda_F \approx p/(8(1+t))$.

Amplification factor as $\Delta t \to \infty$

Methods with small amplication factor as $\Delta t \to \infty$ perform well as large step-sizes.

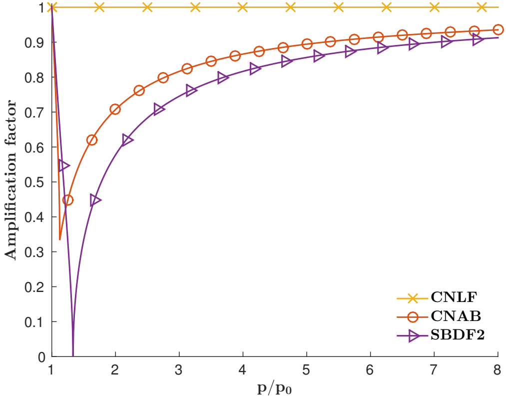

Figure : Amplification factor as $\Delta t \to \infty$.

Numerical Experiments

We consider two classes of problems to demonstrate the effectiveness of our new methods:

1. Image inpainting.
2. Mean curvature motion.

Image Inpainting

In Schönlieb and Bertozzi (2011), the authors proposed the fourth order inpainting model

$$u_t = -\Delta\nabla \cdot \left(\frac{\nabla u}{\sqrt{|\nabla u|^2 + \epsilon^2}} \right) + \lambda_0(u_0 - u),$$

and numerical solution by the first order accurate method

$$\frac{u^{n+1} - u^n}{\Delta t} = -\Delta\nabla \cdot \left(\frac{\nabla u^n}{\sqrt{|\nabla u^n|^2 + \epsilon^2}} \right) + \lambda_0(u_0 - u^n)$$
$$+ p_1\Delta^2 u^n - p_1\Delta^2 u^{n+1} + p_0\lambda u^n - p_0\lambda u^{n+1}.$$

Image inpainting

Figure : TV-H^{-1} image restoration.

Table : Iteration counts for TV-H^{-1} image restoration.

	Δt	Iterations
IMEX Euler	0.30	1002
SBDF2	0.54	401
CNAB	0.64	347

Motion by mean curvature

We evolve the level set equation for motion by mean curvature,

$$u_t = \kappa\,|\nabla u| = |\nabla u|\,\nabla\cdot\left(\frac{\nabla u}{|\nabla u|}\right)$$

on an initial dumbbell-shaped curve in 3D. We solve to time $T = 0.75$ on a $256 \times 128 \times 128$ periodic grid over $(-5,5)\times(-2,2)\times(-2,2)$.

Motion by mean curvature

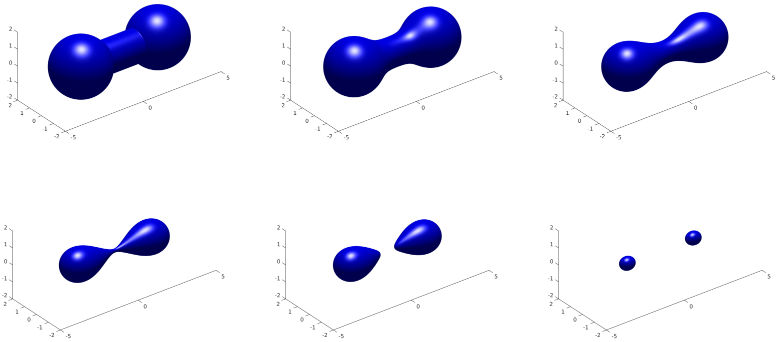

Figure : Mean curvature flow of a dumbbell-shaped curve in 3D. From the left to right, top to bottom, the plots show the evolution at times $t = 0$, 0.10, 0.30, 0.525, 0.55, 0.75.

Motion by mean curvature

On a machine with an Intel® Core™ i5-4570 CPU@3.20GHz running MATLAB 2014b:

- Forward Euler: 3000 time steps $\rightarrow$ over 28 minutes.
- SBDF2: 175 time steps $\rightarrow$ under 2 minutes.

Conclusion

Contributions of this thesis mainly fall into two categories:

1. Further developed linearly stabilized schemes.
   - Outlined a framework for developing new methods of this type.
   - Identified properties necessary for effective schemes.
2. Proposed new methods that outperform existing ones.
   - IMEX multistep methods and exponential time differencing methods.

Future work:

1. Development higher order methods without the deficiencies exhibited by ETD methods.
2. A comparison with popular algorithms for nonlinear stiff PDEs, particularly for image inpainting.