Developing and Analyzing Staggered Designer Multistep Methods

Using MATLAB and Mathematica

Caleb Cramer

Gonzaga University

December 16, 2024

Outline

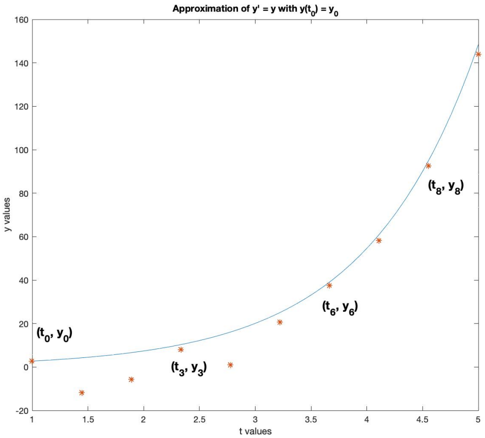
Goal

Approximate solutions to

$$y' = f(t, y), \quad y(t_0) = y_0$$

by finding a sequence:

- $y_k \approx y(t_k)$
  [function values]
- $t_k = t_0 + kh$
  [discrete time grid]
- $f_k = f(t_k, y_k)$
  [derivative values]



Definitions

Definition (Linear Multistep Method)

A linear multistep method utilizes previous data points to approximate the location of a future point. A general k step method can be written as:

$$\begin{aligned} y_{n+1} = & a_1 y_n + a_2 y_{n-1} + \cdots + a_k y_{n+1-k} \\ & + h\left[b_0 f_{n+1} + b_1 f_n + \cdots + b_k f_{n+1-k}\right] \end{aligned}$$

Definition (Order)

The order p of a given method describes the error: $E \approx ch^p$. For example, in a second-order method, we as we decrease the step size by a factor of ten, we would expect the error to decrease by a factor of 10^2.

Definitions

Definition (Stability Domain)

- A region in the complex plane showing when a given method will be stable (roundoff error does not grow).
- Depends on the stepsize h used and the ODE through λ.
- Different for each method
- Use the test problem

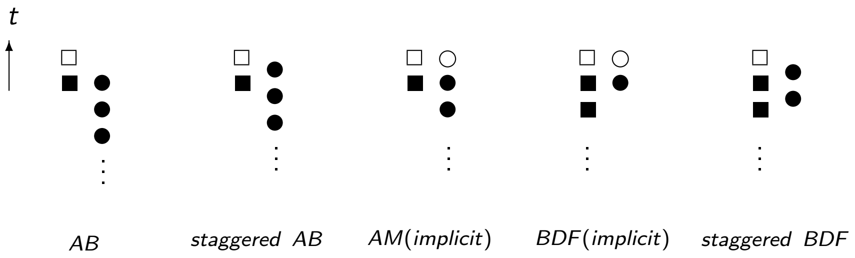
$$y' = \lambda y, \quad y(t_0) = y_0,$$

which has solution $y(t) = y_0 e^{\lambda t}$.

Introduction

Novel way to visualize a method:

- A square represents a y value at a given discrete t
- A circle represents $f = y'$ at a given t



Foundations

Theorem (Dahlquist's Stability Barrier (nonstaggered))

The order p of a k-step stable method must satisfy

$$p \leq \begin{cases} k & \textit{explicit methods} \\ k+1 & k \textit{ is odd and method is implicit} \\ k+2 & k \textit{ is even and method is implicit} \end{cases}$$

Theorem (Ghrist's Extension for Staggered Methods)

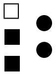
The order p of an explicit k-step stable method must satisfy

$$p \leq \begin{cases} k & k \textit{ is even} \\ k+1/2 & k \textit{ is half-integer} \\ k+1 & k \textit{ is odd} \end{cases}$$

Process for new stencil

- Find the series expansion of the method's error about $h = 0$.
- Maximize the order of the method by setting as many terms equal to 0 as possible (considering previous theorem).
- Solve the corresponding system of linear equations, leaving free parameter(s) as needed.
- Find the domain of the free parameter(s) that give a stable method. Using this domain:
  - Plot the boundary of the stability domain using complex variables and solving difference equations
  - Compare stability domains across the values of the free parameter(s).

Problem 1



1. Expand $error = y(t_{n+1}) - y_{n+1}$
$$\begin{aligned}
&= y(t_{n+1}) - (a_1 y_n + a_2 y_{n-1} + h(b_1 f_{n+1/2} + b_2 f_{n-1/2})) \\
&= (1 - a_1 - a_2)\, y(t_n) + (1 + a_2 - b_1 - b_2)\, hy'(t_n) \\
&\qquad + (1 - a_2 - b_1 + b_2)\, y''(t_n) \frac{h^2}{2} + O(h^3)
\end{aligned}$$
2. Solve the resulting linear system, with free variable a_2
   - $a_1 = 1 - a_2, \quad a_2 = a_2, \quad b_1 = 1, \quad b_2 = a_2$
3. With these values, find the error term (first non-zero term)
   - **Local Error:** $\frac{1+a_2}{24} y'''(\xi_n) h^3$

$$\textbf{Global Error: } \frac{1+a_2}{24} y'''(\xi) \left(t_{final} - t_0\right) h^2$$

$$y_{n+1} = (1 - a_2) y_n + a_2 y_{n-1} + h(f_{n+1/2} + a_2 f_{n-1/2})$$

We apply theory to find this method is stable for $-1 < a_2 \leq 1$.

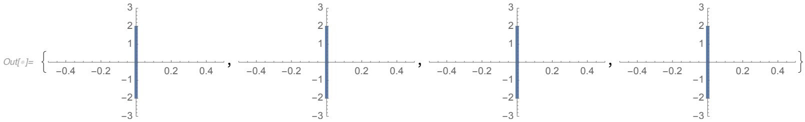


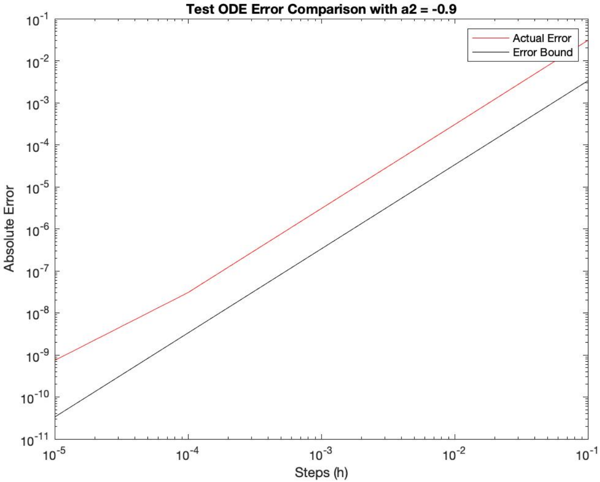
Figure: Stability Domains for Problem 1 for $-0.9 < a_2 \leq 0.9$

Test ODE

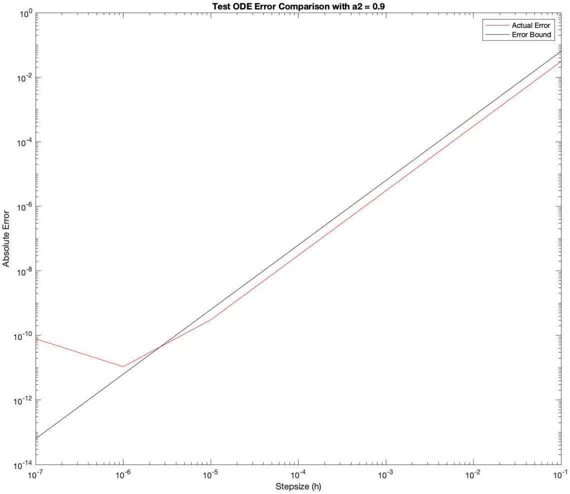
The previous error equation seems to indicate that we can force the error to zero by choosing a_2 values closer and closer to -1. We pursued this idea with the test ODE
$y'' = -4y, y(0) = 1, y'(0) = 0.$

We have proven that the method actually performs like a second-order method.

Test ODE


Test ODE Error Comparison with a2 = -0.9
Actual Error
Error Bound
Absolute Error
Steps (h)
10^-1
10^-2
10^-3
10^-4
10^-5
10^-6
10^-7
10^-8
10^-9
10^-10
10^-11
10^-5
10^-4
10^-3
10^-2
10^-1

Test ODE


Test ODE Error Comparison with a2 = 0.9
Actual Error
Error Bound
Absolute Error
Stepsize (h)
10^0
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-14
10^-7
10^-6
10^-5
10^-4
10^-3
10^-2
10^-1

Problem 2

$$\begin{aligned} y_{n+1} =& (1-a_2)y_n + a_2 y_{n-1} \\ &+ \frac{h}{24}\left[(25+a_2)f_{n+3/2} + (-2+22a_2)f_{n+1/2} + (1+a_2)f_{n-1/2}\right] \end{aligned}$$

Global Error: $\frac{1}{24}y^{(4)}(\xi)\left(t_{final}-t_0\right)h^3$

We apply theory to find this method is stable for $-1 < a_2 \leq 1$.

Problem 2

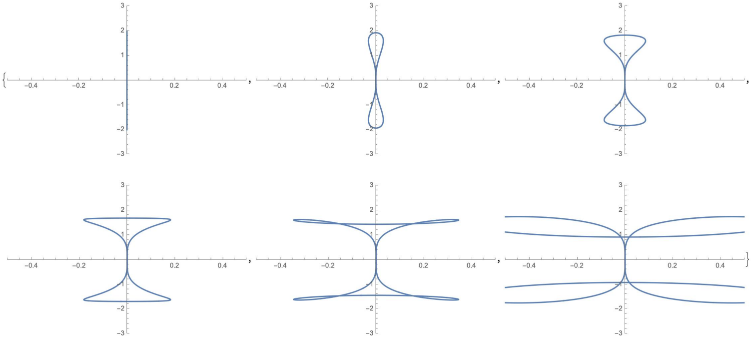
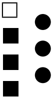


Figure: Stability Domains for Problem 2 for $-0.9 < a_2 \leq 0.9$

$$\text{ISB formula} = \frac{12i(-1+a_2)}{-7+5a_2}$$

Problem 3



- $a_1 = 27 - 24b_3$
- $a_2 = -27 + 24b_3$
- $a_3 = 1$
- $b_1 = b_3$
- $b_2 = -24 + 22b_3$
- $b_3 = b_3$

Global Error:
$\frac{1}{480}(-27 + 17b_3)y^{(5)}(t_{final} - t_0)h^4$

Work in Progress

1. Extending the test ODE to more problems.
2. Can we drive the error to 0 for an actual ODE using the first method (Problem 1)?

Results

1. Problem 1
   - Stability domain proof using complex variables to prove the stability domain is always constant
   - Explored all possible combinations for a_2
   - Tested with against actual ODE
2. Problem 2
   - Explored all possible combinations for a_2
   - Stability domain analysis

Future Work

- Analysis of error on other problems
- Application of test ODE to P3, P4, etc
- Continue adding more terms and higher order with more parameters (Problem 4 and Problem 5)

Acknowledgements

Many thanks to Dr Ghrist for advising and providing much of the basis for this work as well as to the McDonald family for funding this work. And thank you for coming.

Caleb Cramer
caleb_cramer@yahoo.com

Questions

Why would we use these methods over another type?

1. Similar to Runge-Kutta but faster and cheaper
2. Same computational cost for same number of steps but tighter region
3. Better error sometimes
4. Extend further on the imaginary axis

On what types of ODE's would we use these?

- Oscillatory motion problems such as sound, light, etc