Ordinary Differential Equations (I)

Marcin Chrzaszcz, Danny van Dyk
mchrzasz@cern.ch, danny.van.dyk@gmail.com

University of Zurich UZH

Numerical Methods,
14. November, 2016

Announcement

Lecture Cancellation

28. November

Replacement

This Wednesday

16. November

Plan for today

- Catching up with last time
  - What does an implicit Runge-Kutta method look like?

- So far we discussed ODEs of first order
  - How to turn an ODE of order n into a system of n ODEs of first order?

- Specific problem of a given ODE in a boundary value problem
  - How can we use our knowledge of initial value problems to solve these?
  - Can we solve these directly, w/o resorting to initial value problems?

Implicit Runge-Kutta methods

- same as before, method of stage s and order p

$$k_i = f\left[x_n + c_i h, y_n + h \sum_j a_{i,j} k_j\right]$$

- compare Butcher Tableau, which is now fully populated

$$\begin{array}{c|ccccc}
c_1 & a_{1,1} & a_{1,2} & \dots & a_{1,s-1} & a_{1,s} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
c_s & a_{s,1} & a_{s,2} & \dots & a_{s,s-1} & a_{s,s} \\
\hline
 & b_1 & b_2 & \dots & b_{s-1} & b_s
\end{array}$$

- while explicit methods have strictly $s \geq p$, implicit method can achieve orders $p > s$

Gauss-Legendre methods . . .

. . . a subclass of implicit Runge-Kutta methods

Revisit the integral we need:

$$y_{n+1} = y_n + \int_{x_n}^{x_n+h} \mathrm{d}x\, f(x, y(x))$$

- reuse what we know about Gauss quadratures
- rescale the problem such that Legendre polynomials are the appropriate orthogonal basis for $f(x, y(x))$
- example: Gauss Legendre method of stage 2:

$$\begin{array}{c|cc}
\frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\
\frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}$$

- practical aspects of implicit Runge-Kutta methods requires a tangent

Tangent: Fixed-point problems

What is a fixed-point problem?

- consider an automorphism f, e.g.: a function $\mathbb{R} \ni z \mapsto f(z) \in \mathbb{R}$
- a point z^* is a fixed point under f if it fulfills

$$f(z^*) = z^*$$

- A fixed-point problem is finding the fixed point z^* for a given f.
- Iteratively, we can approach $\lim_{n\to\infty} z_n = z^*$ via:

$$z^{(i+1)} = f(z^{(i)})$$

- this works, as long as z^* is an attractive fixed point, i.e.:

$$||f(z^{(i+1)}) - f(z^{(i)})|| < ||z^{(i+1)} - z^{(i)}||$$

- for further reading see the Banach fix-point theorem!

Practical Aspects

We can view implicit single step methods

$$y' = f(x, y) \qquad y_{n+1} = y_n + \Phi(F; x_n, h; y_n, y_{n+1})$$

as a fixed point problem:

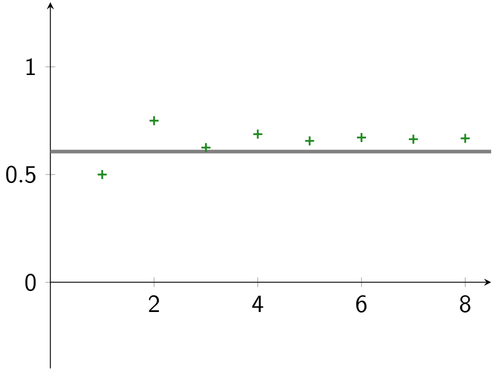
- start with a guess for $y_{n,i=0}$, e.g. based on an explicit method
- compute the new approximation of the implict result for y_{n+1} as

$$y_{n+1,i+1} = y_n + \Phi(F; x_n, h; y_n, y_{n+1,i})$$

- stop after a fixed number of iterations, or when the difference between previous and current iteration falls below a pre-determined threshold.
- check attraction requirement in every step

Example

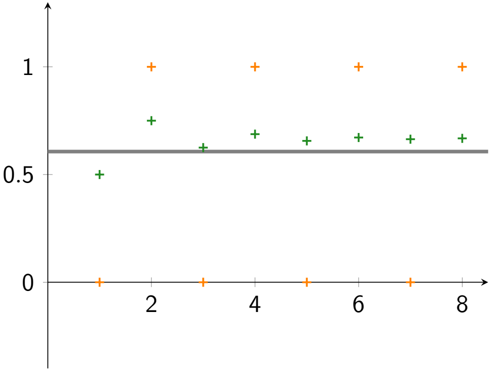
- first step: Euler explicit
- next steps: repeat Euler implicit until $|z^{(i+1)} - z^{(i)}| < 2 \cdot 10^{-3}$



- works fine for $f(x, y) = -y$

Example

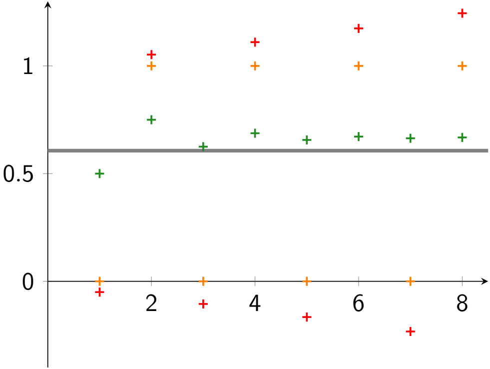
- first step: Euler explicit
- next steps: repeat Euler implicit until $|z^{(i+1)} - z^{(i)}| < 2 \cdot 10^{-3}$



- works fine for $f(x, y) = -y$
- fails for $f(x, y) = -2y$

Example

- first step: Euler explicit
- next steps: repeat Euler implicit until $|z^{(i+1)} - z^{(i)}| < 2 \cdot 10^{-3}$



- works fine for $f(x,y) = -y$
- fails for $f(x,y) = -2y$
- fails spectacularly for $f(x,y) = -2.1y$

Be careful with your starting value!

Toward Boundary Problems: ODEs of order 2

Let's start with an explicit ODE of order 2:

$$y'' = f(x; y, y')$$

Let's rename $y(x) \to z_1(x)$ introduce a new function $z_2(x) = y'(x)$, such that we can write

$$\begin{aligned} z_2' &= f_2(x; z_1, z_2) \qquad && [\equiv f(x; y = z_1, y' = z_2)] \\ z_1' &= f_1(x; z_1) \qquad && [\equiv z_2] \end{aligned}$$

We can now solve the (vector-valued) ODE of order 1.

Generalization to order N

- we can apply this now to any order N ODE to obtain a system of N coupled ODEs of order 1
- however, a first-order system of N equations requires N initial conditions to solve!
- consequently, we already needed N initial conditions to solve the order N system!

Example

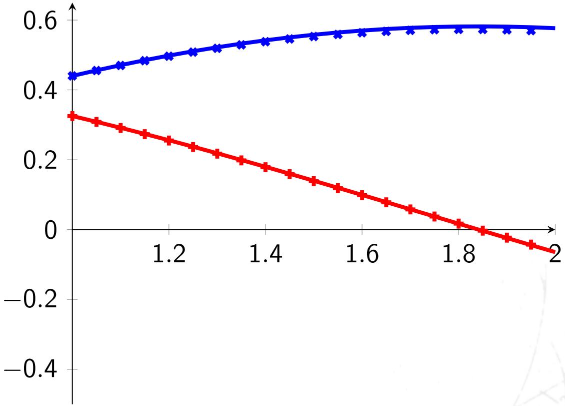
Let's take the Bessel equation:

$$x^2y'' + xy' + (x^2 - \nu^2)y = 0$$

Example

Let's take the Bessel equation: in explicit form

$$y'' = -\frac{1}{x}y' - \left(1 - \frac{\nu^2}{x^2}\right) y$$



IVP

$$y(1) = 0.44051$$
$$y'(1) = 0.325147$$

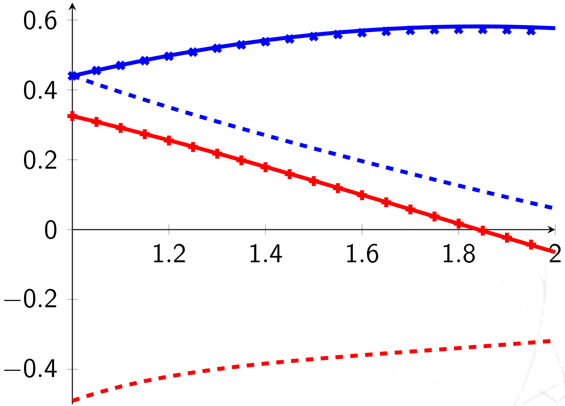
step size: $h = 0.05$

Using implicit Euler method!

Example

Let's take the Bessel equation:

$$y'' = -\frac{1}{x}y' - \left(1 - \frac{\nu^2}{x^2}\right) y$$



IVP (dashed)

$$y(1) = 0.44051$$
$$y'(1) = -0.490276$$

step size: $h = 0.05$

Using implicit Euler method!

Boundary Value Problem and the Shooting Method

- a boundary value problem for a second-order ODE consists of finding the solution to the ODE on the interval $[a, b]$ with the following condition

$$y(a) = \alpha \qquad\qquad y(b) = \beta$$

- we can rewrite it as an initial value problem:

$$y(a) = \alpha \qquad\qquad y'(a) = \gamma$$

- We can the proceed to solve the IVP as a function of γ, such $y(b) = \beta$ is fulfilled.
- This is called the shooting method
- Practically, we define a root-finding problem:

$$0 = F(\gamma) \equiv \beta - y(x = b; \alpha, \gamma)$$

Example

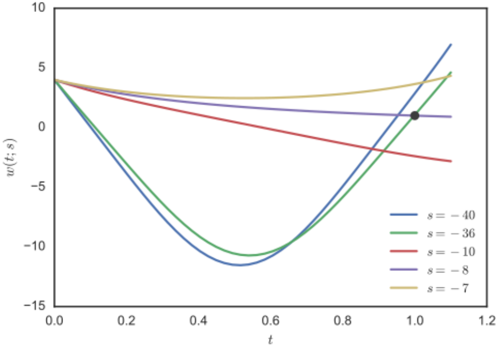
From Stoer, Bulirsch:

$$w''(t) = \frac{3}{2} w^2(t)$$

with boundary conditions

$$w(0) = 4 \qquad w(1) = 1$$

$w'(t) = s$



Example

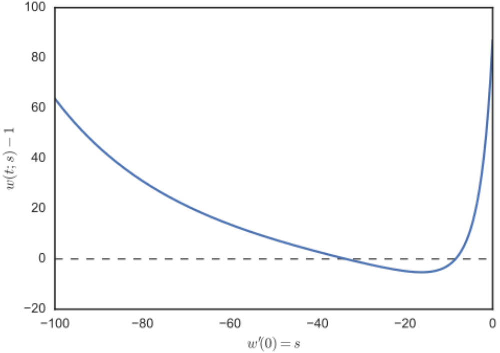
From Stoer, Bulirsch:

$$w''(t) = \frac{3}{2} w^2(t)$$

with boundary conditions

$$w(0) = 4 \qquad\qquad w(1) = 1$$

$w'(t) = s$



two solutions:

1. $w'(0) = -8$
2. $w'(0) \simeq -36.$

Finite Difference Method

- totally different approach then previous ones!
- do not integrate the right-hand side of the ODE!
- instead, discretize the problem (with $k = 1, \ldots, K$)

$$x_k = x_0 + kh \qquad\qquad y_k : \text{independent variables}$$

- discretize the derivatives:

$$y'(x_k) = \frac{y_{k+1} - y_{k-1}}{2h} \qquad y''(x_k) = \frac{y_{k+1} + y_{k-1} + 2y_k}{h^2} \qquad \ldots$$

- boundary value problem now boils down to solving the matrix-valued equation

$$A \cdot \vec{y} = \vec{b}$$

- A only depends on the ODE!
- $y^T = (y_1, \ldots, y_K)$
- b^T to be determined, includes boundary problem in b_1 and b_K

Finite Difference Method

Setup:

- if the ODE does not depend on x, then the matrix A can be prepared early
- in that case, A only depends on the type of differential operator acting on y
- the size K of the problem can easily be adjusted at run time
- the inverse of A can be determined at compile time

$\Rightarrow$ FDMs can be implemented rather efficiently for "easy" differential equations

Convergence:

$$|y(x_k) - y_k| \leq \frac{y^{(4)}(\xi)h^2}{24}(x_k - a)(b - x_k)$$

for $\xi \in [a, b]$.

Finite Difference Method

Example $K = 9,\ h = 1/(K+1) = 0.1$

$$y'' = 4 \qquad y(0) = y_0 = -1 \qquad y(1) = y_{10} = +1$$

The analytic solution reads $y(x) = 2x^2 - 1$

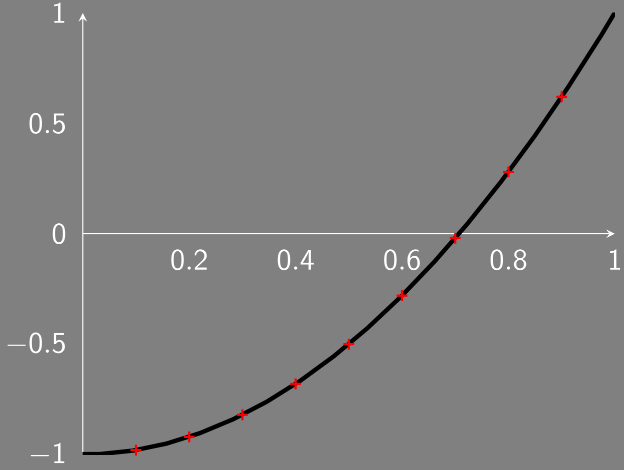
$$\frac{1}{h^2}\begin{pmatrix} -2 & +1 & 0 & \dots & 0 \\ +1 & -2 & +1 & \ddots & \vdots \\ 0 & +1 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & +1 \\ 0 & \dots & 0 & +1 & -2 \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{K-1} \\ y_K \end{pmatrix} = \begin{pmatrix} 4 + \frac{1}{h^2} \\ 4 \\ \vdots \\ 4 \\ 4 - \frac{1}{h^2} \end{pmatrix}$$

Finite Difference Method

Example $K = 9,\ h = 1/(K+1) = 0.1$

$$y'' = 4 \qquad y(0) = y_0 = -1 \qquad y(1) = y_{10} = +1$$

The analytic solution reads $y(x) = 2x^2 - 1$



Backup