MAT 186 Module D4: Euler's Method
Section LEC 0108

Adam Morgan
adam.morgan@mail.utoronto.ca

October 20, 2023

Plan for Today

1. Brief Review/Discussion of ODE Modules
2. Warm-Up Problem
3. Interpretive problems, discussion in groups
4. Code demos

ODEs: The Story So Far

- We defined ODEs, & showed how to check a given function solves an ODE.
- Some ODEs cannot be solved with pen and paper: MUST be done numerically (remember the van der Pol oscillator + at the end of today I'll give another example).
- Computed equilibrium solutions to ODEs.
- Obtained qualitative information on solutions from phase lines and slope fields (ie. seeing if solutions tend to a particular equilibrium over time).
- Existence-Uniqueness Theorem: gives conditions guaranteeing that a given ODE can actually be solved.
- PCE D4 + today's material: how to obtain numerical solutions to ODEs yourself! Use linear approximation = Euler's method

ODEs: The Story So Far

- We defined ODEs, & showed how to check a given function solves an ODE.
- Some ODEs cannot be solved with pen and paper: MUST be done numerically (remember the van der Pol oscillator + at the end of today I'll give another example).
- Computed equilibrium solutions to ODEs.
- Obtained qualitative information on solutions from phase lines and slope fields (ie. seeing if solutions tend to a particular equilibrium over time).
- Existence-Uniqueness Theorem: gives conditions guaranteeing that a given ODE can actually be solved.
- PCE D4 + today's material: how to obtain numerical solutions to ODEs yourself! Use linear approximation = Euler's method

ODEs: The Story So Far

- We defined ODEs, & showed how to check a given function solves an ODE.
- Some ODEs cannot be solved with pen and paper: MUST be done numerically (remember the van der Pol oscillator + at the end of today I'll give another example).
- Computed equilibrium solutions to ODEs.
- Obtained qualitative information on solutions from phase lines and slope fields (ie. seeing if solutions tend to a particular equilibrium over time).
- Existence-Uniqueness Theorem: gives conditions guaranteeing that a given ODE can actually be solved.
- PCE D4 + today's material: how to obtain numerical solutions to ODEs yourself! Use linear approximation = Euler's method

Here are some ODEs you will almost certainly end up solving numerically in your career:

1. Nonlinear oscillators (mechanics, electrical circuits),

$$\ddot{\mathbf{x}}(t) + \omega^2 \mathbf{x}(t) = \mathbf{F}\left(t, \mathbf{x}(t), \dot{\mathbf{x}}(t)\right)$$

2. Gradient descent equation (optimization, optimal control, machine learning),

$$\dot{\mathbf{x}}(t) = -\nabla J\left(\mathbf{x}(t)\right)$$

3. Stochastic ODEs (mechanics, chemistry, finance),

$$\mathrm{d}\mathbf{x} = \mathbf{F}\left(t, \mathbf{x}(t)\right)\mathrm{d}t + \sigma\left(t, \mathbf{x}(t)\right)\mathrm{d}\mathbf{W},$$

where $\mathrm{d}\mathbf{W}$ is “white noise”.

... to say nothing of the **partial** *differential equations you'll encounter!*

Differential Equations in Engineering Practice

Here are some ODEs you will almost certainly end up solving numerically in your career:

1. Nonlinear oscillators (mechanics, electrical circuits),

$$\ddot{\mathbf{x}}(t) + \omega^2 \mathbf{x}(t) = \mathbf{F}\left(t, \mathbf{x}(t), \dot{\mathbf{x}}(t)\right)$$

2. Gradient descent equation (optimization, optimal control, machine learning),

$$\dot{\mathbf{x}}(t) = -\nabla J\left(\mathbf{x}(t)\right)$$

3. Stochastic ODEs (mechanics, chemistry, finance),

$$\mathrm{d}\mathbf{x} = \mathbf{F}\left(t, \mathbf{x}(t)\right) \mathrm{d}t + \sigma\left(t, \mathbf{x}(t)\right) \mathrm{d}\mathbf{W},$$

where $\mathrm{d}\mathbf{W}$ is "white noise".

... to say nothing of the **partial** *differential equations you'll encounter!*

Here are some ODEs you will almost certainly end up solving numerically in your career:

1. Nonlinear oscillators (mechanics, electrical circuits),

$$\ddot{\mathbf{x}}(t) + \omega^2 \mathbf{x}(t) = \mathbf{F}\left(t, \mathbf{x}(t), \dot{\mathbf{x}}(t)\right)$$

2. Gradient descent equation (optimization, optimal control, machine learning),

$$\dot{\mathbf{x}}(t) = -\nabla J\left(\mathbf{x}(t)\right)$$

3. Stochastic ODEs (mechanics, chemistry, finance),

$$\mathrm{d}\mathbf{x} = \mathbf{F}\left(t, \mathbf{x}(t)\right)\mathrm{d}t + \sigma\left(t, \mathbf{x}(t)\right)\mathrm{d}\mathbf{W},$$

where $\mathrm{d}\mathbf{W}$ is "white noise".

... to say nothing of the **partial** *differential equations you'll encounter!*

Euler's Method

(Forward) Euler Approximation

Consider the ODE

$$x'(t) = f(t, x(t)).$$

Suppose we have a decent estimate of $x(t)$ for some t. For a small, fixed $\Delta t > 0$ and nice enough $f(x)$, we then have an approximation of $x(t + \Delta t)$:

$$x(t + \Delta t) \approx x(t) + \Delta t \; f(t, x(t)).$$

- Derived from linear approximation of $x(t)$ at t.
- Can successively apply Euler's method to approximate $x(s)$ for $s \gg t$: first compute $x(t + \Delta t)$, then $x(t + 2\Delta t), \ldots$
- Approximation error is directly proportional to Δt, *most of the time* (more on this later).
- Most of today: understanding how good Euler does on problems *where we have an exact soln to compare to.*

Euler's Method

(Forward) Euler Approximation

Consider the ODE

$$x'(t) = f(t, x(t)).$$

Suppose we have a decent estimate of $x(t)$ for some t. For a small, fixed $\Delta t > 0$ and nice enough $f(x)$, we then have an approximation of $x(t + \Delta t)$:

$$x(t + \Delta t) \approx x(t) + \Delta t \ f(t, x(t)).$$

- Derived from linear approximation of $x(t)$ at t.
- Can successively apply Euler's method to approximate $x(s)$ for $s \gg t$: first compute $x(t + \Delta t)$, then $x(t + 2\Delta t), \ldots$
- Approximation error is directly proportional to Δt, *most of the time* (more on this later).
- Most of today: understanding how good Euler does on problems *where we have an exact soln to compare to.*

Warm-Up Problem

Problem

Consider the ODE

$$x'(t) = f(x(t)).$$

Suppose this equation has a single equilibrium, x_*.

- Pick any $\Delta t > 0$. Starting from $x(0) = x_*$, perform a single step of Euler's method to obtain an approximation $x(\Delta t)$.
- How accurate is your approximation? Does your answer depend on Δt?
- Do you agree with the statement "Euler's method preserves equilibria"?

Solution to Warm-Up Problem

Interpretive Problem 1

Problem

1. Use Euler's method with a step size of $\Delta t = 0.1$ to approximate the solution of

$$\begin{cases} x'(t) = 3t^2 x(t) \\ x(0.5) = 1. \end{cases}$$

at time $t = 0.6$.

2. Verify that

$$x(t) = e^{t^3 - 0.5^3}$$

solves the IVP above. Then, compute the absolute error in the Euler approximation of $x(0.6)$ you found in part 1.

3. Obtain an estimate of $x(0.6)$ that is twice as accurate as the one from part 1.

Solution to Interpretive Problem 1

Solution to Interpretive Problem 1

Interpretive Problem 2, Part 1

Problem

Consider the IVP

$$\begin{cases} x'(t) = -15x(t) \\ x(0) = 1. \end{cases}$$

Use Euler's method with a time-step size of $\Delta t = \frac{1}{8}$ to estimate

$$x\left(\frac{1}{8}\right),\ x\left(\frac{1}{4}\right),\ x\left(\frac{3}{8}\right),\ \text{and } x\left(\frac{1}{2}\right).$$

What do you think about the quality of your approximation?

Solution to Interpretive Problem 2, Part 1

Solution to Interpretive Problem 2, Part 1

Interpretive Problem 2, Part 2

Problem

1. Use linear approximation to write $x(t)$ in terms of $x\left(t+\Delta t\right)$ and $x'\left(t+\Delta t\right)$.
2. Use the previous item to derive the **backward Euler method** for numerically solving $x'(t) = f(t, x(t))$:

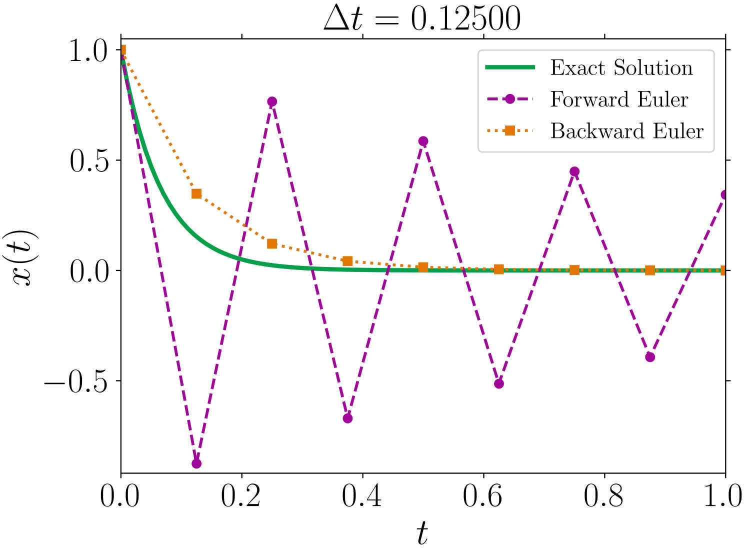
$$x\left(t+\Delta t\right) \approx x(t) + \Delta t\ f\left(t+\Delta t, x\left(t+\Delta t\right)\right).$$

3. Re-do the last problem using the backward Euler method instead. Does the quality of approximation change?

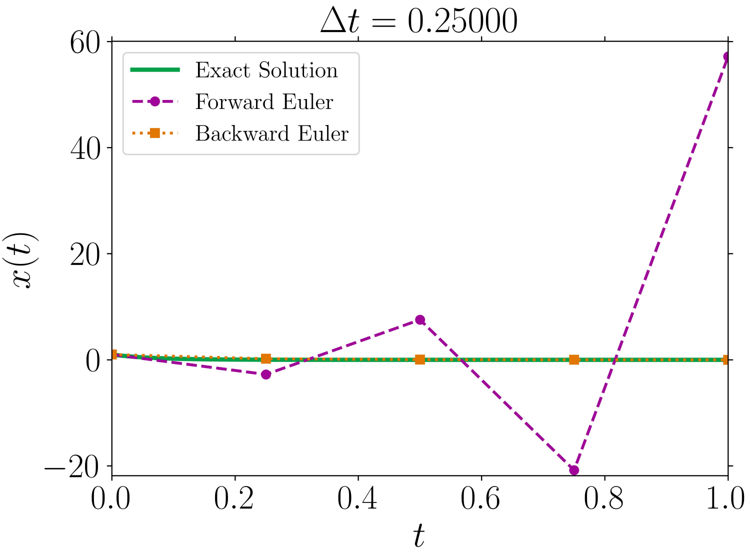
Solution to Interpretive Problem 2, Part 2

Solution to Interpretive Problem 2, Part 2

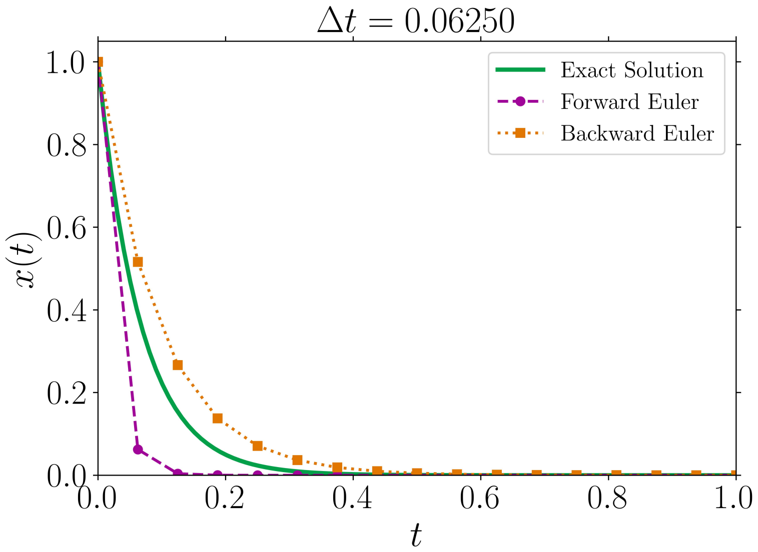
Forward vs. Backward Euler



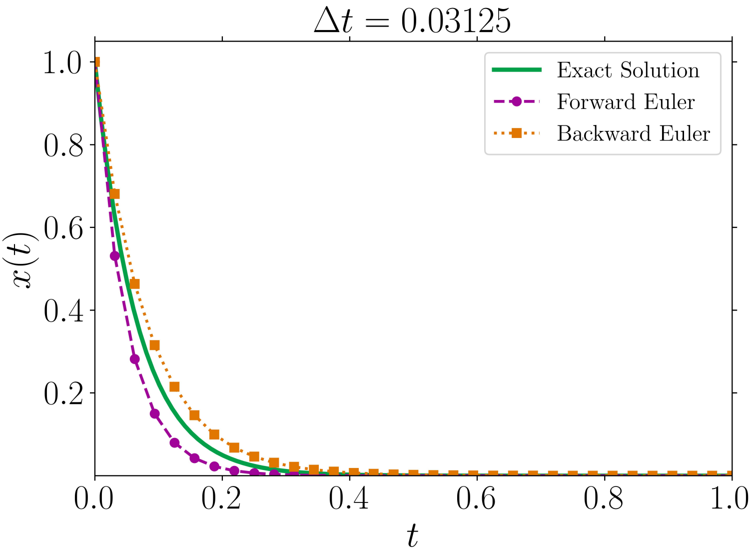
Forward vs. Backward Euler



Forward vs. Backward Euler

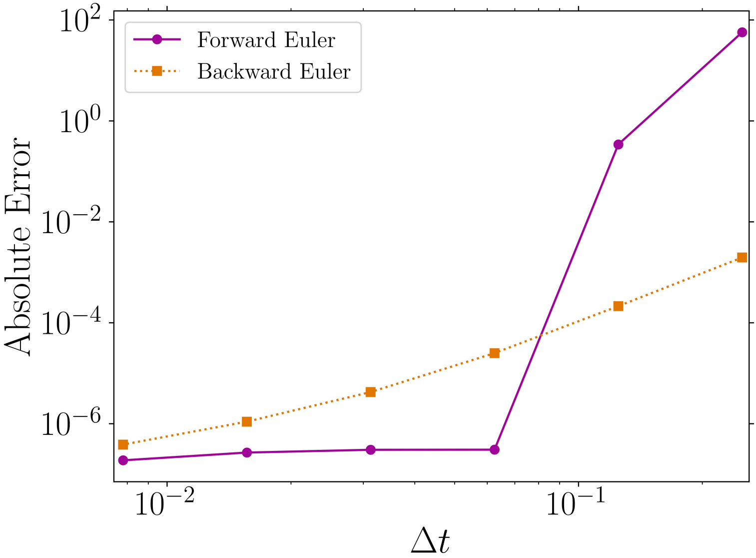


Forward vs. Backward Euler



Forward vs. Backward Euler

Look at error in approximating $x(1)$...



Takeaways from this Problem

- While (forward) Euler's method does a decent job most of the time, *sometimes* it's not so hot
- If Euler fails, there are other methods one can try!
- The failure of Euler's method is sometimes called **stiffness** (defn. is a bit imprecise), a type of **numerical instability**.
- Strictly speaking, forward Euler worked well if the time step Δt was sufficiently small. Question for discussion: why might such a restriction be an issue in practice?

Takeaways from this Problem

- While (forward) Euler's method does a decent job most of the time, *sometimes* it's not so hot
- If Euler fails, there are other methods one can try!
- The failure of Euler's method is sometimes called **stiffness** (defn. is a bit imprecise), a type of **numerical instability**.
- Strictly speaking, forward Euler worked well if the time step Δt was sufficiently small. Question for discussion: why might such a restriction be an issue in practice?

Takeaways from this Problem

- While (forward) Euler's method does a decent job most of the time, *sometimes* it's not so hot
- If Euler fails, there are other methods one can try!
- The failure of Euler's method is sometimes called **stiffness** (defn. is a bit imprecise), a type of **numerical instability**.
- Strictly speaking, forward Euler worked well if the time step Δt was sufficiently small. Question for discussion: why might such a restriction be an issue in practice?

Code Demos

To the Jupyter notebook!