Numerical Solution of ODE IVPs *

L.G. de Pillis and A.E. Radunskaya

July 30, 2002

*This work was supported in part by a grant from the W.M. Keck Foundation

NUMERICAL SOLUTION OF ODE IVPs

Overview

1. Quick review of direction fields.
2. A reminder about _____(1) and _____(2).
3. Important test: Is the ODE initial value problem _____(3)?
4. Fundamental concepts: Euler's Method.
5. Fundamental concepts: Truncation error.
6. Fundamental concepts: _____(4) of a method.
7. Fundamental concepts: _____(5) of a method.
8. Stiff ODEs.
9. Other methods overview.
10. Systems and higher order IVPs.
11. Solving IVPs with packaged software.

Numerical Solution of ODE IVPs

Notes for Overview slide:

Answers:
(1) existence
(2) uniqueness
(3) well-posed
(4) Order
(5) Stability

We are assuming the students have had an introductory course in ordinary differential equations, so they have seen direction fields and existence and uniqueness theory before. Direction fields are re-introduced to the students because they provide a natural lead-in to the geometric derivation of Euler's method. As you discuss the items on the Overview list, you may want to keep in mind the following:

- Euler's method is introduced for illustration only. It is a good pedagogical device for giving understanding about how numerical ODE solvers work fundamentally,

but it is a method that nobody should use in practice. There are far better methods available for use.

- Euler's method is an example of an "explicit" "single-step" method.
- When we introduce other methods, we will not get into any details at all, since those can be learned in a course on numerical analysis. We will simply give an overview of the categories of solvers available.

Numerical Solution of ODE IVPs

Direction Field Review

General First Order Ordinary Differential Equation:

$$y' = f(t, y)$$

- y' is shorthand for _____(1).
- $f(t, y)$ is a function of the _____(2) variable t and the _____(3) variable y.

Assumptions:

- $f(t, y)$ is defined and single valued in some rectangular region R in the $t - y$ plane.
- If $y = y(t)$ is a solution, then it is differentiable at all points in R. This allows us to plot a smooth curve.

Numerical Solution of ODE IVPs

Notes for Direction Field slide:

Answers:

(1) dy/dt

(2) independent

(3) dependent

This is just a review, the students should have seen these before.
Note that the initial conditions are not yet specified.

Numerical Solution of ODE IVPs

Direction Field Review – Demo

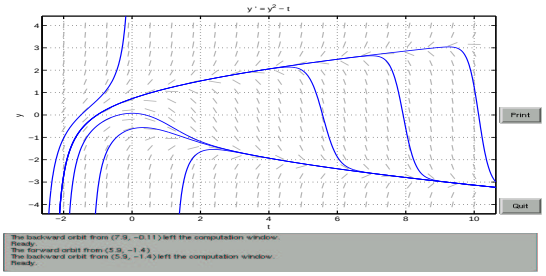
A direction field should be plotted _____(1).

Step 1: Draw a region in the $t - y$ plane.

Step 2: Choose a point (a, b) in the region.

Step 3: Plot a short line starting at (a, b) with slope $f(a, b)$.

Step 4: Repeat steps (2) and (3) for many different points (a, b).



Numerical Solution of ODE IVPs

Notes for Direction Field Demo slide:

Answers:

(1) numerically, *i.e.,* on the computer

This is a plot of the direction field for $y' = y^2 - t$, with $-2 < t < 10$ and $-4 < y < 4$. A few solutions are also plotted.

This plot was generated in MATLAB 6 using the Rice University code called dfield6, currently available at

http://math.rice.edu/~dfield

Another demo program that could be used is ODEArchitect, either the Windows based version, or the newer Java web browser version. Maple and Mathematica also have the appropriate capabilities.

Point out: The slope of the ODE solution at any point (a, b) in the plane is $f(a, b)$. Plotting the slope at that point tells us where the ODE solution is going next. We can plot many of these direction lines all through the region. Once the direction field is plotted, it is possible to guess at the approximate path of one particular solution to the ODE in

the family of solutions, since any solution is tangent to these direction field lines. Any single solution is then specified by a point that it goes through.

As a preview, the following questions could be asked:

1. Given a point (a, b), is there always a solution through that point tangent to the direction field? *If the direction field is "vertical", y' is not defined at (a, b).*

2. Given a point (a, b), is there only one solution that is tangent to the direction field (a, b)? *The answer in general is "no", but for the systems we will study (those that satisfy existence and uniqueness theorems) the answer is "yes".*

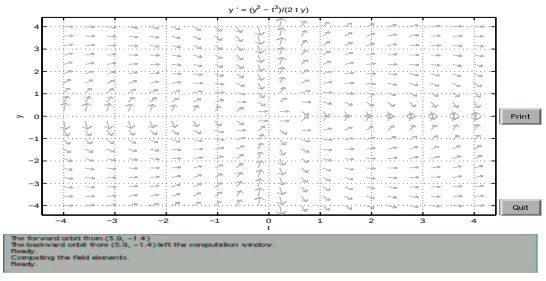
Numerical Solution of ODE IVPs

Direction Field Review – Notes

- A direction field gives a sense of the _____(1) of the solutions.
- Warning: be careful plotting a line that has a _____(2) (dividing by zero).

  Example:

$$y' = (y^2 - t^2)/(2ty)$$



Numerical Solution of ODE IVPs

Notes for Direction Field Review Notes slide:

Answers:

(1) flow

(2) vertical slope

The ODE example has a singularity at $y = 0$ and at $t = 0$. Try making a direction field for $-4 \leq t \leq 4$ and $-4 \leq y \leq 4$. Try this in any demo package you like. You'll find that you can create the direction field, but if you try to draw the solution near to the singularity, the generation of the solution path may get stuck.

Numerical Solution of ODE IVPs

Existence and Uniqueness: A Reminder

Questions we must ask:

- Is there a solution?
- Is there only one solution?

Why are these questions important?

1. If there is _____(1), your computer program may _____(2).
2. If there is _____(3), the program will _____(4). It may not be the one you want.

From your ODEs course, you learned theorems that gave checklists ensuring the existence and uniqueness of ODE IVP solutions.

A TIP: Use these theorems.

Numerical Solution of ODE IVPs

Notes for Existence and Uniqueness A Reminder slide:

Answers:

(1) no solution

(2) still produce output

(3) more than one solution

(4) choose a solution for you

The students should already have been exposed to the theorems regarding existence and uniqueness of solutions. If you feel it is appropriate, you may provide handouts that summarize these theorems. You may wish to point out to the students that the more serious computing we do, the more we *must* rely on fundamental mathematical theory to inform us about the validity of our computed solutions.

Possible Example: You may wish find an example for which the computational output is incorrect, and the theory shows that you are likely to get into trouble (*e.g.*, if there are multiple solutions). For example, $y' = 3y^{2/3}$ (or use any power less than 1) and initial data $y(0) = 0$. This has $y(t) = 0$ as a solution, but also $y(t) = t^3$ is a solution. What would a numerical solver do with this problem?

Numerical Solution of ODE IVPs

Well-Posed Problems: A Must for Computation

Before you begin computing a solution to an IVP, you must ensure that

- the problem is _____(1)

This automatically ensures that the solution to the problem

- exists
- is unique

Basic meaning: The solution of a *well-posed problem* is not only unique, but also is

_____(2) to

_____(3)

in the data (which almost always will occur on a computer).

Notes for Well-Posed Problems A Must for Computation slide:

Answers:

(1) well-posed

(2) not too sensitive

(3) small perturbations

Note: To make computation of an accurate solution feasible, not only must an ODE IVP have a solution that exists and is unique, but the problem must also be well-posed. A mathematical problem is well-posed if in addition to existence and uniqueness, the solution also depends continuously on the problem data. This will be formally defined on the next slide.

Well-posedness is very important because if the solution to the perturbed ODE is very different from the unperturbed solution, it is very difficult to get a good computational answer. So, for the students, well-posedness is a necessity.

In reality, well-posedness of a problem is considered highly desirable, but is not always achievable. For example, the problem of determining an illness based on symptoms is not well posed, since the same set of symptoms (e.g., stomach ache) can be

the result of more than one cause (food poisoning, viral infection, bacterial infection, etc.). Heath [Hea02, p. 3] points out that inferring the internal structure of a physical system solely from external observations , as in tomography or seismology, often leads to mathematical problems that are inherently ill-posed in that distinctly different internal configurations may have indistinguishable external appearances.

Numerical Solution of ODE IVPs

Well-Posed Problems: Formal Definition

Definition: The IVP

$$y' = f(t, y), \;\; a \leq t \leq b, \;\; y(a) = y_0$$

is a *well-posed* problem if

(1) A _____(1) solution $y(t)$ to the problem _____(2).

(2) A number $\epsilon > 0$ exists such that a _____(3) solution $z(t)$ to the _____(4)

$$z' = f(t, z) + \delta(t), \;\; a \leq t \leq b, \;\; z(a) = y_0 + \epsilon_0$$

exists whenever _____(5) and _____(6)

(3) A constant $\kappa > 0$ exists with the property that

$$|z(t) - y(t)| < \kappa\epsilon \quad \text{for all } \; t \in [a, b]$$

Numerical Solution of ODE IVPs

Notes for Well-Posed Problems Formal Definition slide:

Answers:

(1) unique
(2) exists
(3) unique
(4) perturbed problem
(5) $|\epsilon_0| < \epsilon$
(6) $|\delta(t)| < \epsilon$ for all $t \in [a, b]$

This definition is found in [BFR81, p.184].

Part (1) of the definition simply guarantees the existence of a unique solution to the original IVP. Part (2) of the definition guarantees the existence of a unique solution to the perturbed IVP. Part (3) of the definition says that the original solution and the perturbed solution are very close.

Numerical Solution of ODE IVPs

Well-Posed Problems: An Easy Test?

Question: How can we tell whether criteria $(1) - (3)$ for well-posedness are satisfied for a particular problem?

Answer: Good news! There is an *easy test* (*i.e.*, a theorem) that tells us immediately whether an IVP is well-posed.

Numerical Solution of ODE IVPs

Well-Posed Problems: An Easy Test!

Theorem: Suppose $D = \{(t, y) | t \in [a, b] \text{ and } y \in [c, d]\}$. The IVP

$$\frac{dy}{dt} = f(t, y), \;\; t \in [a, b], \;\; y(a) = y_0$$

is well-posed provided

1. f is _____(1) on D
2. f satisfies a _____(2) in the variable _____(3) on the set _____(4)

Notes for Well-Posed Prolems An Easy Test slide:

Answers:

(1) continuous
(2) Lipschitz Condition
(3) y
(4) D

This test for well-posedness is found in [BFR81, p.184].

- These conditions are sufficient conditions.
- With this theorem, we only need to test the right-hand-side $f(t, y)$ of the ODE, we do not need to find any a priori solutions.
- The next slide explains what a Lipschitz condition is. The students may not have seen this before.

Comments from [Gea71, p.7]: In many problems we cannot get a Lipschitz condition for all y, but only in a region of the y-space. [For example, if $f(t, y) = y^2$, $\partial f/\partial y$

exists and is bounded in any finite region.] The proof of this Lipschitz theorem is valid as long as both z and y remain in the region, so it may be necessary to limit the maximum perturbation ϵ. For example, perturbations to $y' = 1/y^2,\ y(0) > 0$, are bounded as long as the perturbation does not reduce y below 0. Consequently, it is well posed with respect to any positive initial value y_0, but when y_0 is close to 0, ϵ is small and κ is large.

Numerical Solution of ODE IVPs

Lipschitz Continuity: Definition

Definition: A function $f(t, y)$ is said to be *Lipschitz continuous* or to satisfy a *Lipschitz condition* in the variable y on a set $D \subset \Re^2$ provided a constant $L > 0$ exists with the property that

$$|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2|$$

whenever $(t, y_1), (t, y_2) \in D$. The constant L is called the *Lipschitz constant*.

Note: If $f(t, y)$ is differentiable, then the Lipschitz condition guarantees that _____(1). Conversely, if f is differentiable with respect to y and _____(2), then f satisfies the Lipschitz condition. This property can be used as a _____(3) of whether the _____(4) is satisfied.

Numerical Solution of ODE IVPs

Notes for Lipschitz slide:

Answers:

(1) $|\partial f/\partial y| \leq L$

(2) $|\partial f/\partial y| \leq L$

(3) test

(4) Lipschitz condition

Numerical Solution of ODE IVPs

Lipschitz Condition Test: Example

Example: Determine whether the IVP $y' = f(t, y),\ \ y(t_0) = y_0$ on D is well posed, given

$$f(t, y) = ty$$

and

$$D = \{(t, y) | t \in [1, 2],\ y \in [-3, 4]\}$$

If so, find the Lipschitz constant.

Answer: For each $(t, y_1),\ \ (t, y_2)$ in D, we have

$$|f(t, y_1) - f(t, y_2)| = |ty_1 - ty_2| = t|y_1 - y_2| \leq 2|y_1 - y_2|$$

So, $f(t, y)$ is _____(1) in _____(2) on _____(3), with Lipschitz constant $L =$ _____(4).

Numerical Solution of ODE IVPs

Notes for Lipschitz Example slide:

Answers:

(1) Lipschitz continuous

(2) y

(3) D

(4) 2

The example was borrowed from [BFR81, p. 182].

Numerical Solution of ODE IVPs

Well-Posed versus Stable Solutions (1)

Question: How much can a perturbed solution $\hat{\vec{\mathbf{y}}}(t)$ with perturbed $\hat{\vec{\mathbf{f}}}(t, \hat{\vec{\mathbf{y}}})$ and perturbed initial conditions $\hat{\vec{\mathbf{y}}}(t_0) = \hat{\vec{\mathbf{y}}}_0$ _____(1) from the original solution $\vec{\mathbf{y}}(t)$ with original $\vec{\mathbf{f}}(t, \vec{\mathbf{y}})$ and original initial conditions $\vec{\mathbf{y}}(t_0) = \vec{\mathbf{y}}_0$ when _____(2) continuity with constant L on a bounded domain D is assumed?

Answer:

$$||\hat{\vec{\mathbf{y}}}(t) - \vec{\mathbf{y}}(t)|| \leq e^{L(t-t_0)}||\hat{\vec{\mathbf{y}}}_0 - \vec{\mathbf{y}}_0|| + \frac{e^{L(t-t_0)} - 1}{L}||\hat{\vec{\mathbf{f}}} - \vec{\mathbf{f}}||$$

where $||\hat{\vec{\mathbf{f}}} - \vec{\mathbf{f}}|| = \max_{(t,\vec{\mathbf{y}})\in D} ||\hat{\vec{\mathbf{f}}}(t, \vec{\mathbf{y}}) - \vec{\mathbf{f}}(t, \vec{\mathbf{y}})||$.

Numerical Solution of ODE IVPs

Notes for Well-Posed versus Stable (1) slide:

Answers:

(1) differ

(2) Lipschitz

Note: The bound on the difference between the original solution and the perturbed solution is derived in [Hea02, p.388]. It is not necessary to go into detail here. However, it may be useful to point out to the students that the first additive portion of the bound is due to perturbations in the initial data, while the second additive portion is due to perturbations in the function $\vec{\mathbf{f}}$.

Note: Even a well-posed problem may have perturbed solutions that diverge exponentially over time. Therefore, we also talk about *solutions* of ODEs that are *stable* and *asymptotically stable*, for which perturbed solutions do not diverge by more than a constant amount over time. Make sure to point out to the students that the word “stable” is commonly used both to refer to the sensitivity of solutions of an ODE and to refer to the sensitivity of a numerical algorithm. So we can have a “stable solution”, and we can have a “stable algorithm”. This can initially be confusing to students. However, they

are likely to see these two uses for the word “stable” in textbooks and other literature. Stability of *numerical algorithms* will be discussed in later slides.

Note: **This slide and the next slide on the stability of an ODE solution may be skipped if you wish**.

Numerical Solution of ODE IVPs

Well-Posed versus Stable Solutions (2)

Definition: A solution of the ODE $\vec{\mathbf{y}}' = \vec{\mathbf{f}}(t, \vec{\mathbf{y}})$ is stable if for every $\epsilon > 0$ there is a $\delta > 0$ such that if $\hat{\vec{\mathbf{y}}}(t)$ satisfies the ODE and $||\hat{\vec{\mathbf{y}}}(t_0) - \vec{\mathbf{y}}(t_0)|| \leq \delta$ then $||\hat{\vec{\mathbf{y}}}(t) - \vec{\mathbf{y}}(t)|| \leq \epsilon$ for all $t \geq t_0$.

Example: The solution of the IVP $y' = \lambda y$ with $y(t_0) = y_0$ and λ a constant, is given by $y(t) = y_0 e^{\lambda t}$. So,

- If $\lambda > 0$, every solution is _____(1).
- If $\lambda < 0$, every solution is _____(2).

Notes for Well-Posed versus Stable (2) slide:

Answers:

(1) unstable

(2) stable

Notes: See [Hea02] for more on this definition of stable, as well as a discussion about how to determine whether an ODE solution is stable.

This definition of stability of an ODE solution says that if the initial value is perturbed then the perturbed solution remains within some constant range of the original solution. Therefore, an ODE may be well-posed but not stable. Stability of the ODE solution is also desirable when computing solutions. However, good packaged solvers can generally handle well-posed but unstable solutions with little problem.

Note also: A stable solution is said to be asymptotically stable if $||\hat{\vec{y}}(t) - \vec{y}(t)|| \to 0$ as $t \to \infty$. So for the example, when $\lambda < 0$, every solution is even asymptotically stable.

You may wish to ask the students if the solutions are stable when $\lambda = 0$. The solutions are, of course, stable, since they are constant. However, they are not asymptotically stable. In the module on Qualitative Analysis, we use these same terms when discussing the stability of equilibria, that is, the stability of constant solutions.

Numerical Solution of ODE IVPs

Fundamental Concepts: Euler's Method (1)

Disclaimer: Never use _____(1) actually to _____(2) an IVP. It is introduced here only to _____(3) basic concepts and definitions.

Our canonical IVP:

$$\frac{dy}{dt} = f(t, y), \;\; y(t_0) = y_0$$

on some region D in the plane.

Assumption: All problems we will see are well-posed.

Numerical Solution of ODE IVPs

Notes for Fundamenta Concepts Euler's Method (1) slide:

Answers:

(1) Euler's method

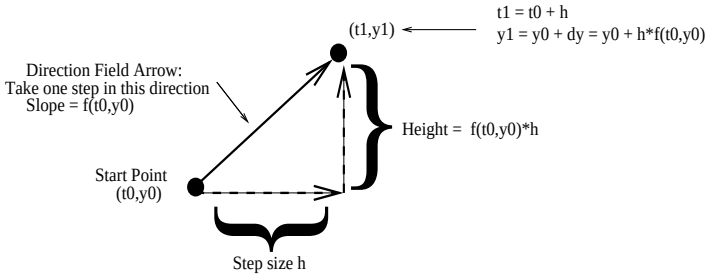
(2) solve

(3) illustrate

We are assuming students have also seen Euler's method, so this is just a review. We derive it from the geometric perspective on the next slide.

Numerical Solution of ODE IVPs

Fundamental Concepts: Euler's Method (2)



In general: Starting at (t_0, y_0), we get to (t_{n+1}, y_{n+1}) by using

$$\begin{aligned} t_{n+1} &= t_n + h \\ y_{n+1} &= y_n + hf(t_n, y_n) \end{aligned}$$

Note: y_{n+1} represents the *numerical approximation* to $y(t_{n+1})$.

Euler's method is: _____(1) and _____(2).

Numerical Solution of ODE IVPs

Notes for Fundamental Concepts Euler's Method (2) slide:

Answers:

(1) single-step

(2) explicit

Note:

- A "multi-step" method includes more "history". That is, one would see functions of t_n as well as t_{n-1}, t_{n-2}, etc. on the right-hand-side of the ODE.
- An "implicit" method would have a function of (t_{n+1}, y_{n+1}) on the right-hand-side of the ODE.

Suggestion: You may want to insert a demo here (in MATLAB, for example) of an implementation of Euler's method. One good example to try:

$$\begin{aligned} y' &= -y + t + 1, \ t \in [0, 1] \\ y(0) &= 1 \end{aligned}$$

Actual solution is $y(t) = t + e^{-t}$. Try step size $h = 0.1$, so $t_n = 0.1n$. This example is borrowed from [BFR81, p.188]. Euler's method works fine for this simple example. MATLAB demo code: See *NumDemo1* scripts.

Numerical Solution of ODE IVPs

Truncation Error

Truncation Error (TE) arises from the _____(1) of the true solution.

Usually, a _____(2) or _____(3) summation approximates an _____(4).

Example: Taylor expand $y(t_{n+1})$ to get Euler's Method (a type of "Taylor Method"):

$$\underbrace{y(t_{n+1}) = y(t_n) + hy'(t_n)}_{\text{Euler's Method: Truncated Series}} + \underbrace{\frac{h^2}{2}y''(t_n) + \mathcal{O}(h^3)}_{\text{Local Truncation Error (LTE)}}$$

Numerical Solution of ODE IVPs

Notes for Truncation Error slide:

Answers:

(1) mathematical approximation
(2) finite
(3) truncated
(4) infinite series

Using the Taylor expansion of $y(t_{n+1})$ is an alternate way to derive Euler's method. Truncation Error (TE) is the error that arises because of the mathematical approximation to the actual solution. TE usually refers to error arising from using a truncated or finite summation to approximate the sum of an infinite series. In the case of Euler's method, the infinite series is a Taylor expansion. Euler's method is considered a type of "Taylor Method". Euler's method truncates the Taylor series after two terms. Higher order Taylor methods truncate the series after more terms have been included. Note that we are implicitly assuming that $y(t)$ is analytic, that is, that it has a Taylor series expansion.

Numerical Solution of ODE IVPs

Order of a Method

Generally, Local Truncation Error (LTE) can be approximated by $\alpha h^{\mathbf{k}}$ in the sense that

$$\lim_{h\to 0}\left(\frac{\text{LTE}(h)}{h^{\mathbf{k}}}\right) = \alpha$$

In general: The number $k-1$ is the order of the numerical method.

Example: Let y_1 be the numerical approximation to $y(t_1)$. Then LTE in Euler's method is:

$$y(t_1) - y_1 = \frac{h^{\mathbf{2}}}{2}y''(t_0) + \cdots = \alpha h^{\mathbf{2}}$$

Here, $k =$ _____(1), so Euler's method is *order* _____(2).

Terminology:

An *order* _____(3) method implies:

The _____(4) **error behaves like** _____(5).

Notes for Order of a Method slide:

Answers:

(1) 2
(2) 1
(3) n
(4) accumulated
(5) h^n

The "order of a method" reveals the *global* or *accumulated* error in a method. The Local Truncation Error LTE depends on the step-size h and the method begin used. For Euler's method, for example, LTE shrinks like $\mathcal{O}(h^2)$ as h shrinks, and Euler's method is "order 1". The larger n is (which changes with the method) the better. The global or accumulated error behaves like h in the case of Euler. So, if we were to halve h, our global error would be halved (not counting rounding). For an order 2 method, halving h would cut our global error to $1/4$ its previous size.

Numerical Solution of ODE IVPs

Summary: Types of Numerical Error

- Rounding error: From finite precision floating point arithmetic.
  (Example: $\frac{1}{3} \approx 0.3333$)
- Truncation error: From the method used. Two classes:
  - **Local (LTE)**: Error made in _____(1) of the numerical method.
  - **Global (GTE)**: Accumulated error. Error made after _____(2) of the numerical method.
- A numerical method is "order n" if LTE = $\mathcal{O}(h^{n+1})$.

Notes for Types of Numerical Error slide:

Answers:

(1) one step

(2) several steps

"Truncation error" is also sometimes referred to as "Discretization error". TE arises from the method used, and would remain even if the floating point arithmetic were perfect.

LTE: $y(t_1) - y_1$ where $y(t_1)$ is the true solution of the ODE passing through the previous point (t_0, y_0).

GTE: $e_n = y(t_n) - y_n$ where $y(t_n)$ is the true solution of the ODE passing through the initial point (t_0, y_0).

Note: An exercise to try: Confirming the order of a method. See *e.g.* [Dan85, p. 21]. Also try with second or fourth order Runge-Kutta.

Numerical Solution of ODE IVPs

Stability of a Numerical Method: Introductory Example

Idea: If _____(1) do not cause the _____(2) solution to diverge from the _____(3) solution, the numerical method is stable.

Example: Given the test IVP

$$y' = \lambda y, \;\; y(0) = y_0$$

apply Euler's method with step size _____(4):

$$y_{n+1} = y_n + h\lambda y_n = (1 + h\lambda) y_n$$

which implies that

$$y_{n+1} = \underbrace{(1 + h\lambda)}_{\text{Amplification Factor}}^{n+1} y_0$$

Notes for Stability:Introductory Example slide:

Answers:

(1) small perturbations

(2) numerical

(3) true

(4) h

Essentially: If small perturbations do not cause the resulting numerical solution to diverge away without bound, the method is considered stable.

The example problem given in this slide is well-posed. This is the canonical IVP on which stability of numerical methods is often tested. If a method is stable on this problem, it is likely to be stable on more complicated problems. The exact solution to the IVP is $y(t) = y_0 e^{\lambda t}$.

Numerical Solution of ODE IVPs

Stability of a Numerical Method: Introductory Example (2)

Example (continued): The "Growth" or "Amplification Factor" = $(1 + h\lambda)$.

Therefore,

- If $|1 + h\lambda| \leq 1$, Euler's method is _____(1).
- If $|1 + h\lambda| > 1$, Euler's method is _____(2).

Requirements for Euler to be stable for this example:

- If λ **complex**: $h\lambda$ must be inside a _____(3) disk in the complex plane centered at _____(4).
- If λ **real**: $h\lambda$ must be in the interval _____(5).

Note: Choice of step-size h is crucial for stability.

Notes for Stability(2) slide:

Answers:

(1) stable
(2) unstable
(3) radius 1
(4) -1
(5) $(-2, 0)$

If Euler is unstable, that means the solution will grow without bound. With Euler's method, the step-size h is the only thing we have control over (λ is determined by the IVP). Therefore, in order to maintain stability, h must often be very small (especially in the case of stiff problems, which we will see soon).

Numerical Solution of ODE IVPs

Stability of a Numerical Method: General System

General System: _____(1)

Taylor Expand: $\vec{\mathbf{y}}(t+h) =$ _____(2)

Let $t = t_k$ in Taylor expansion:

$$\vec{\mathbf{y}}(t_{k+1}) = \vec{\mathbf{y}}(t_k) + h\vec{\mathbf{f}}(t, \vec{\mathbf{y}}) + \mathcal{O}(h^2) \quad (1)$$

Euler's method:

$$\vec{\mathbf{y}}_{k+1} = \vec{\mathbf{y}}_k + h\vec{\mathbf{f}}(t, \vec{\mathbf{y}}) \quad (2)$$

Subtract equation (1) from equation (2):

$$\underbrace{\vec{\mathbf{y}}_{k+1} - \vec{\mathbf{y}}(t_{k+1})}_{\text{Global error } e_{k+1}} = (\vec{\mathbf{y}}_k - \vec{\mathbf{y}}(t_k)) + h \underbrace{\left(\vec{\mathbf{f}}(t_k, \vec{\mathbf{y}}_k) - \vec{\mathbf{f}}(t_k, \vec{\mathbf{y}}(t_k))\right)}_{\text{Apply Mean Value Theorem}} - \mathcal{O}(h^2)$$

Numerical Solution of ODE IVPs

Notes for Stability General System slide:

Answers:

(1) $\vec{\mathbf{y}}' = \vec{\mathbf{f}}(t, \vec{\mathbf{y}})$.

(2) $\vec{\mathbf{y}}(t) + h \underbrace{\vec{\mathbf{y}}'(t)}_{\vec{\mathbf{f}}(t,\vec{\mathbf{y}})} + \mathcal{O}(h^2)$

So what can we say about a more general *system* of ODEs? At this point, we introduce the notion of stability in the context of a system, because the only difference here between a system and a single equation is the notation. All concepts carry over.

Numerical Solution of ODE IVPs

Stability of a Numerical Method: General System (2)

Applying the Mean Value Theorem:

$$\vec{\mathbf{f}}(t_k, \vec{\mathbf{y}}_k) - \vec{\mathbf{f}}(t_k, \vec{\mathbf{y}}(t_k)) = \mathbf{J}_f(t_k, \alpha\vec{\mathbf{y}}_k + (1-\alpha)\vec{\mathbf{y}}(t_k))(\vec{\mathbf{y}}_k - \vec{\mathbf{y}}(t_k))$$

where

- $\mathbf{J}_f =$ _____(1) matrix of $\vec{\mathbf{f}}$ w.r.t. $\vec{\mathbf{y}}$ and $\alpha \in [0, 1]$.

_____(2) is expressed in general as:

$$\vec{\mathbf{e}}_{k+1} = \underbrace{(\mathbf{I} + h\mathbf{J}_f)}_{\text{Amplification Factor}} \vec{\mathbf{e}}_k + \text{LTE}_{k+1}$$

Requirement for _____(3):

$$\rho(\mathbf{I} + h\mathbf{J}) \leq 1$$

where ρ represents the _____(4) of a matrix.

Notes for Stability General System (2) slide:

Answers:

(1) Jacobian

(2) Global Error

(3) stability

(4) spectral radius Note: Recall for your students the Definition: The spectral radius of a matrix $\mathbf{A}$ is the maximum $|\lambda|$, where λ is an eigenvalue of $\mathbf{A}$.

The Global Error is multiplied at each step of the numerical method by $(\mathbf{I} + h\mathbf{J})$ As long as the spectral radius of this matrix is ≤ 1, errors won't grow unboundedly and the method is considered stable.

Numerical Solution of ODE IVPs

Stability of a Numerical Method: General System (3)

- Observation: Stability requires that $\rho(\mathbf{I} + h\mathbf{J}) \leq 1$.
- Question: What does this stability restriction imply?
- Answer: All eigenvalues of $h\mathbf{J}_f$ must lie inside a _____(1) disk in the _____(2) centered at _____(3).
- Note: If eigenvalues lie _____(4) the disk, the method will be _____(5).
- Implication: We must choose _____(6) so that all stability constraints are satisfied.

Notes for Stability General System (3) slide:

Answers:

(1) radius 1

(2) complex plane

(3) -1

(4) outside

(5) unstable

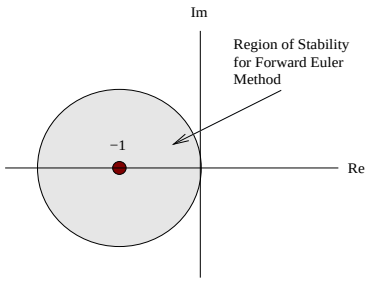
(6) step size h

Numerical Solution of ODE IVPs

Stability of a Numerical Method:

Euler Method Region of Stability

All eigenvalues of $h\mathbf{J}_f$ must lie inside the disk.



Numerical Solution of ODE IVPs

Stability of a Numerical Method: Euler Example

Example: Consider

$$y' = -10(t-1)y, \;\; y(0) = e^{-5}, \;\; t \in [0, 2]$$

Question: When will Euler's Method be stable?

Answer: Notice that EM implies

$$y_{k+1} = y_k + h(-10(t-1)y_k) = \underbrace{(1 - 10h(t-1))}_{\text{Amplification Factor}} y_k$$

Therefore,

- For _____(1), the method is unstable for any _____(2).
- For _____(3), the method will be stable if _____(4).

Numerical Solution of ODE IVPs

Notes for Stability - Euler Example slide:

Answers:

(1) $t < 1$

(2) h

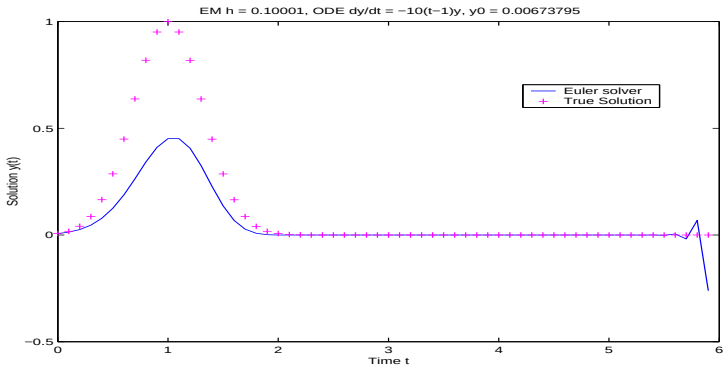
(3) $t > 1$

(4) $h < \frac{1}{5}\left(\frac{1}{t-1}\right)$

Example borrowed from [KMN89, p.289]. Note that in this example, h is required to get smaller and smaller as t grows. Therefore, for a fixed h, we can expect the method to become unstable after a certain point in time.

Numerical Solution of ODE IVPs

Stability of a Numerical Method: Euler Example Illustration



Numerical Solution of ODE IVPs

Notes for Euler Example Illustration slide:

The exact solution to this IVP is $y(t) = e^{-5(t-1)^2}$.

You may wish to run a demo in class in which you actually solve the IVP from this example with different step sizes, and see when the solution becomes unstable. The illustration inclued here shows the solution is inaccurate before $t = 1$, and also goes visibly unstable around $t = 5$ with stepsize $h = 1.0001$. MATLAB demo code: See *NumDemo2* scripts.

Numerical Solution of ODE IVPs

Implicit Methods

- Recall: Euler's method (EM) is _____(1) and _____(2). The limited region of stability for EM requires we choose _____(3) carefully.
- Improvement: Make the _____(4) _____(5).
- How? Use information at _____(6) as wells as at _____(7).
- This makes the method _____(8).

Numerical Solution of ODE IVPs

Notes for Implicit Methods slide:

Answers:

(1) explicit
(2) single-step
(3) step size h
(4) region of stability
(5) larger
(6) t_{k+1}
(7) t_k
(8) implicit

Implicit methods tend to be more stable. Recall: Euler's method is explicit and single step. But the region of stability is limited, and one must be careful to choose step-size h properly. One solution to this limitation is to make the region of stability larger by using information at t_{k+1} as well as at t_k. This makes the method *implicit*.

Numerical Solution of ODE IVPs

Implicit Methods: Backward Euler

- Example of an Implicit Method: *Backward Euler Method (BE)*

$$\vec{\mathbf{y}}_{k+1} = \vec{\mathbf{y}}_k + h\vec{\mathbf{f}}(\underbrace{t_{k+1}, \vec{\mathbf{y}}_{k+1}}_{\text{Implicit}})$$

- Question: How can we solve for $\vec{\mathbf{y}}_{k+1}$?
- Answer:
  - Use _____(1). This often requires calculating the _____(2) of the function $\vec{\mathbf{f}}(t, \vec{\mathbf{y}})$.
  - Use _____(3) methods.
- Note: Both approaches require an initial guess, usually derived by taking one step of an explicit method.

Numerical Solution of ODE IVPs

Notes for Implicit Methods BE slide:

Answers:

(1) root-finding

(2) derivative or Jacobian

(3) predictor-corrector

The example we provide is the Backward Euler (BE) method. This introduces the additional complication of figuring out how to solve a possibly nonlinear equation for its root. The question about solving for $\vec{\mathbf{y}}_{k+1}$ is important, especially if the right hand side $\vec{\mathbf{f}}(t, \vec{\mathbf{y}})$ of the ODE is nonlinear in $\vec{\mathbf{y}}$. One can use a built in root finder, or write ones own code. Using a built-in root finder is a good way to go, if the root finder is given a sufficiently close initial guess. However, how many iterations a root-finder will need per time step may become large in some cases. It is also common to implement Predictor-Corrector methods in this case. These are very easy to write and implement, and may take fewer steps per iteration, but one must be very careful of issues involving convergence of the method.

Numerical Solution of ODE IVPs

Implicit Methods: Stability

- Trade-off: Implicit methods take _____(1) but are more _____(2).
- Apply BE to the test ODE $y' = \lambda y$:

$$\begin{aligned} y_{k+1} &= y_k + h\lambda y_{k+1} \\ \text{Therefore} \Rightarrow y_k &= \underbrace{\left(\frac{1}{1-h\lambda}\right)^k}_{\text{Amplification Factor}} y_0 \end{aligned}$$

- For BE to be stable we require

$$\left|\frac{1}{1-h\lambda}\right| < 1$$

- Good news! As long as $\Re(\lambda) < 0$, BE is stable for any step size h.

Numerical Solution of ODE IVPs

Notes for Implicit Methods: Stability slide:

Answers:

(1) longer

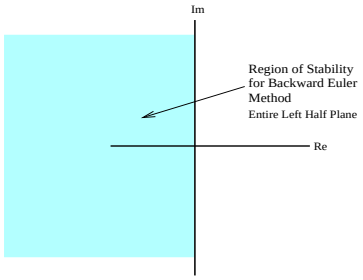
(2) stable

With implicit methods, the solution process is longer (since at each time step, a root-finding iteration is taking place), but stability is much better. In this slide we illustrate the improved stability property of the BE method. Middle step of solve from BE method to final solution is $(1 - h\lambda)y_{k+1} = y_k$.

Numerical Solution of ODE IVPs

Stability of a Numerical Method: BE Stability Region Illustration

All eigenvalues of $h\mathbf{J}_f$ must lie in the left half plane.



Numerical Solution of ODE IVPs

Implicit Methods: Stability Notes

Notes:

- BE is _____(1) accurate (just like _____(2))
- For a _____(3), the stability requirement becomes $\rho((\mathbf{I} - h\mathbf{J}_f)^{-1}) < 1$
- The stability region for BE is the entire _____(4) of the complex plane (as compared to the radius 1 disk of EM).
- Since _____(5) step size h keeps us within the stability region, the BE method is called _____(6).
- There exist higher order implicit methods, like the Trapezoid Method (order = _____(7)).
- Not all implicit methods are _____(8).

Notes for Implicit Methods: Stability Notes slide:

Answers:

(1) 1st order
(2) EM
(3) system
(4) left half
(5) any
(6) unconditionally stable
(7) 2
(8) unconditionally stable

An example of a higher order implicit method that is also unconditionally stable is the Trapezoid method:

$$y_{k+1} = y_k + h(f(t_k, y_k) + f(t_{k+1}, y_{k+1}))/2$$

The Trapezoid Method is second order accurate and unconditionally stable for all eigenvalues in the LHS of the complex plane. The Trapezoid method is also known as 2nd

order Adams-Moulton. Not all implicit methods are unconditionally stable, but they do tend to have larger regions of stability than do explicit methods.

Numerical Solution of ODE IVPs

Stiffness in ODEs

Question: What is "stiffness"?

Answer:

- Physically: A process whose components have _____(1) time scales. Also, a process whose time scale is _____(2) compared to the time interval over which it is being observed.
- Mathematically: A well-posed ODE $\vec{\mathbf{y}}' = \vec{\mathbf{f}}(t, \vec{\mathbf{y}})$ is "stiff" if its Jacobian $\mathbf{J}_f$ has _____(3) that differ greatly in magnitude.
- Practically: An ODE is stiff if an explicit method (like EM) is _____(4) because stability requirements force the step size h to be extremely small.

Numerical Solution of ODE IVPs

Notes for Stiffness in ODEs slide:

Answers:

(1) highly disparate
(2) short
(3) eigenvalues
(4) inefficient

Note: Some of these observations were borrowed from [Hea02, p. 401]. The requirement that step-size h be extremely small for stability of stiff problems is undesirable because it is usually smaller than one would need h to be to achieve reasonable accuracy.

Numerical Solution of ODE IVPs

Stiffness in ODEs: Example 1

Example 1: For test ODE $y' = \lambda y$ with $t \in [a, b]$, the problem is _____(1) if _____(2).

Recall: For stability in EM, we require $|1 - h\lambda| < 1$. If λ is real and $\lambda << -1$, this forces _____(3).

For a **system**: A system of ODEs with $t \in [a, b]$ is _____(4) when _____(5) where λ_j are the _____(6) of Jacobian $\mathbf{J}(t, \vec{\mathbf{y}}(t))$.

Numerical Solution of ODE IVPs

Notes for Stiff ODEs Example 1 slide:

Answers:

(1) stiff
(2) $(b-a)\Re(\lambda) << -1$
(3) h to be very small
(4) stiff
(5) $(b-a)\min_j(\Re(\lambda_j)) << -1$
(6) eigenvalues

Numerical Solution of ODE IVPs

Stiffness in ODEs: Example 2

Example 2: Consider

$$y' = -\alpha(y - \sin t) + \cos t, \;\; y(0) = 1, \;\; t \in [0, 1]$$

Let $\alpha = 1000$. Then the Jacobian is _____(1). So eigenvalue $\lambda =$ _____(2). Therefore,

$$(1 - 0)\Re(\lambda) = (-\alpha) = -1000 << -1$$

This is _____(3) on $t \in [0, 1]$.

Note: on another interval, $t \in [0, 0.002]$ we have

$$(0.002 - 0)\Re(\lambda) = (0.002)(-1000) = -2$$

so this ODE is _____(4) on this interval.

Numerical Solution of ODE IVPs

Notes for Stiff ODEs Example 2 slide:

Answers:

(1) $\frac{\partial f}{\partial y} = -\alpha$

(2) α

(3) stiff

(4) not stiff

Example 2 borrowed from [KMN89, p. 285].

Numerical Solution of ODE IVPs

Stiffness: EM vs BE

Consider

$$y' = -100y + 100t + 101, \;\; y(0) = 1$$

Let $h = 0.1$. Computed output with perturbed initial data:

Time t	0.00	0.10	0.20	0.30	0.40
Exact Soln	$\mathbf{1.00}$	$\mathbf{1.10}$	$\mathbf{1.20}$	$\mathbf{1.30}$	$\mathbf{1.40}$
EM	0.99	1.19	0.39	8.59	-64.21
EM	1.01	1.01	2.01	-5.99	67.01
BE	0.00	1.01	1.19	1.30	1.40
BE	2.00	1.19	1.21	1.30	1.40

Numerical Solution of ODE IVPs

Notes for EM vs BE slide:

This example was borrowed from [Hea02, p. 402]. The general solution to the ODE is $y(t) = 1 + t + ce^{-100t}$. In general, $y(0) = 1 + c$. But with $y(0) = 1$, this implies $c = 0$. The problem is stiff.

We have MATLAB code that solves this problem, both with EM and BE. The data in the table are taken from the MATLAB code, and are confirmed in Heath [Hea02, p.402].

MATLAB demo code: See *NumDemo3* scripts.

Note that we have not addressed stiff problems with rapidly oscillating solutions. The approach there would be different. See suggestions in Stoer and Bulirsch.

Numerical Solution of ODE IVPs

Stiffness: Comments on EM vs BE

Note:

- EM _____(1) with only _____(2) perturbations.
- BE is _____(3) even with _____(4) perturbations.

Numerical Solution of ODE IVPs

Notes for Stiffness:Comments slide:

Answers:

(1) breaks down
(2) small
(3) robust
(4) large

Numerical Solution of ODE IVPs

Stiffness: Summary

- A particular ODE may be _____(1) or _____(2).
- A numerical ODE solving method can be _____(3) or _____(4) for a particular problem.
- The stability of the numerical method often depends on _____(5).
- _____(6) methods should (almost) always be used to solve stiff ODEs.

Numerical Solution of ODE IVPs

Notes for Stiffness:Summary slide:

Answers:

(1) stiff
(2) nonstiff
(3) stable
(4) unstable
(5) the step-size h
(6) Implicit

We say that implicit methods should *almost* always be used to solve stiff ODEs, because the degree of stiffness can vary. It is possible to encounter a somewhat stiff problem that can be solved by using automatic step size adjustment with an explicit method. However, it is usually wisest to stick with the implicit methods when there is stiffness in the ODE.

Numerical Solution of ODE IVPs

Other IVP Solvers

Other classes of numerical IVP solvers include (but are not limited to):

- Higher Order Taylor methods (seldom used)
  - Can give _____(1) accuracy.
  - They require the computation of the _____(2) of $f(t, y)$.
- Runge-Kutta methods (very popular)
  - Can give _____(3) accuracy.
  - Do not need _____(4) of $f(t, y)$.
  - Can be _____(5) or _____(6).
  - These are _____(7)-step methods.
  - Methods include: Midpoint, Modified Euler, Heun, 4th Order Runge-Kutta (RK4)

Numerical Solution of ODE IVPs

Notes for Other IVP Solvers slide:

Answers:

(1) high
(2) derivatives
(3) high
(4) derivatives
(5) explicit
(6) implicit
(7) single

Notes: High order Taylor methods have not been so popular because they involve calculating the derivative of your function. However, in the future, newer automatic differentiation methods may allow for increased use of High order Taylor methods.

Numerical Solution of ODE IVPs

Other IVP Solvers (cont)

- Multi-step methods
  - Can give _____(1) accuracy.
  - Can be _____(2) or _____(3).
  - Starting values must be calculated with a _____(4) method (*e.g.*, RK4)
  - Methods include: Adams-Bashforth, Adams-Moulton, Milne, Simpson
- Extrapolation methods
  - These take solutions generated by lower order methods, and increase _____(5) by _____(6).
  - Variations of these methods presented in [SB93], [Ste73], and [Gra65]. Also see the discussion and reference list in [Asa95, p.642].

Numerical Solution of ODE IVPs

Notes for Other IVP Solvers slide:

Answers:

(1) high
(2) explicit
(3) implicit
(4) single-step
(5) accuracy
(6) extrapolation

Numerical Solution of ODE IVPs

Systems and Higher Order IVPs

- All methods and theories presented can be _____(1) to apply to systems.
- Many higher order IVPs can be _____(2) to 1st order systems of IVPs. Then all methods and theories apply here, too.

Example: Suppose we have the second order equation describing a linear spring,

$$y'' = -ky, \quad y(0) = y_0, \quad y'(0) = v_0$$

Convert this to a 2×2 first order system of equations:

$$\begin{aligned} y' &= v \\ v' &= \text{_____(3)} \end{aligned}$$

Numerical Solution of ODE IVPs

Notes on Systems slide

Answers:

(1) directly generalized

(2) converted

(3) $-ky$

Every ODE solving concept we have discussed to this point can be extended to solving systems of 1st order ODE IVPs. This implies that higher order IVPs that can be converted to a first order system can also be solved with the same techniques and theories. Students should already have seen how to do this kind of conversion in their introductory ODE class. If there is a need to refresh their memory, it may be appropriate to show a couple of simple examples. In sum: Methods to solve 1st order systems of ODEs are simply generalizations of the methods for a single 1st order equation.

Numerical Solution of ODE IVPs

Solving IVPs with Packaged Software

To solve a system of ODE IVPs $\vec{\mathbf{y}}' = \vec{\mathbf{f}}(t, \vec{\mathbf{y}})$ with $y(t_0) = y_0$ with a packaged routine typically requires _____(1) to supply the following:

- The name of the routine that computes $\vec{\mathbf{f}}(t, \vec{\mathbf{y}})$.
- _____(2) and _____(3) values for times t.
- Initial value y_0.

$\cdots$ and for some solvers, sometimes $\cdots$

- The number of equations in the system.
- _____(4) and/or _____(5) error tolerances.

$\cdots$ and sometimes for a stiff ODE $\cdots$

- The routine that computes the Jacobian $\mathbf{J}_f$ of function $\vec{\mathbf{f}}$.

Numerical Solution of ODE IVPs

Notes on Packaged Software slide

Answers:

(1) the user
(2) Initial
(3) final
(4) Absolute
(5) relative

Note: The output for many of these solvers usually includes the solution $\vec{\mathbf{y}}$ at the final time t, but can often include a whole string of solutions $\vec{\mathbf{y}}$ at various times t. Some solvers may also provide warning messages, or measures of the quality of the solution generated.

At this point it might be a good idea to provide some actual codes for the students to look at. For example, in MATLAB, one could provide a file in which the ODE is defined, as well as a file that calls a standard MATLAB solver, like ODE45. This is a good point to assign some exercises, as well. One type of exercise might involve having the students solve various IVPs and IVP systems using some built-in solvers, and then comparing accuracy and speed.

References

[Asa95] N. S. Asaithambi. *Numerical Analysis: Theory and Practice*. Saunders COllege Publishing, 1995.

[BFR81] Richard L. Burden, J. Douglas Faires, and Albert C. Reynolds. *Numerical Analysis*. Prindle, Weber & Schmidt, second edition, 1981.

[Dan85] J. M. A. Danby. *Computing Applications to Differential Equations*. Reston Publishing Company, 1985.

[Gea71] C. William Gear. *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall Series. Prentice-Hall, 1971.

[Gra65] W. Gragg. On extrapolation algorithms for ordinary initial value problems. *SIAM Journal on Numerical Analysis*, 2:384–403, 1965.

[Hea02] Michael T. Heath. *Scientific Computing: An Introductory Survey*. McGraw Hill, second edition, 2002.

[KMN89] David Kahaner, Cleve Moler, and Stephen Nash. *Numerical Methods and Software*. Prentice Hall Series in Computational Mathematics. Prentice-Hall, 1989.

[SB93] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer, second edition, 1993.

[Ste73] H.J. Stetter. Analysis of discretization methods for ordinary differential equations. In *Tracts in Natural Philosophy*. Springer, 1973.