CS537

Numerical Analysis

Lecture

Numerical Solution of Ordinary Differential Equations

Professor Jun Zhang

Department of Computer Science
University of Kentucky
Lexington, KY 40206-0046

April 15, 2010
An Ordinary Differential Equation (ODE) is an equation that involves one or more derivatives of an unknown function.

A solution of a differential equation is a specific function that satisfies the equation.

For the ODE $\frac{dx}{dt} - x = e^t$, the solution is $x(t) = te^t + ce^t$, where $c$ is a constant. Note that the solution is not unique.

Also, the function has derivatives only respect to one variable.

For derivatives with multiple variables, the equation is usually called partial differential equation (PDE).
An Example of ODE

An ODE solution as a function of the time $t$
A differential equation does not, in general, determine a unique solution function. Additional conditions are needed to determine a unique solution.

The **initial-value problem** has the following standard form

\[
\begin{cases}
  x' = f(t, x) \\
  x(a) = s
\end{cases}
\]

Here \( x \) is a function of \( t \), \( x(a) \) is the given initial value of the problem. We would like to determine the value of \( x \) at any time before or after \( a \).

Some examples of initial value problems and solutions:

<table>
<thead>
<tr>
<th>Equation</th>
<th>Initial Value</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x' = x + 1 )</td>
<td>( x(0) = 0 )</td>
<td>( x = e^t - 1 )</td>
</tr>
<tr>
<td>( x' = 6t - 1 )</td>
<td>( x(1) = 6 )</td>
<td>( x = 3t^2 - t + 4 )</td>
</tr>
</tbody>
</table>
Numerical Solution

Analytical solutions in closed forms are only available for special differential equations.

Most differential equations encountered in practical applications do not have analytical solutions.

Some analytical solutions may be too complicated to be useful in practice.

A numerical solution of a differential equation is usually obtained in the form of a table of discrete values.

Interpolation procedures can be used to obtain all values of the approximate numerical solution within a given interval.

A large amount of data may be displayed as a solution curve on a color graphical monitor.
ODE and Integration

Consider the differential equation

\[
\begin{align*}
\frac{dx}{dr} &= f(r, x) \\
x(a) &= s
\end{align*}
\]

We integrate it from \( t \) to \( t+h \)

\[
\int_t^{t+h} dx = \int_t^{t+h} f(r, x(r))dr
\]

We obtain

\[
x(t+h) = x(t) + \int_t^{t+h} f(r, x(r))dr
\]

Replacing the integral with one of the numerical integration rules we studied before, we obtain a formula for solving the differential equation
Numerical Rules

Using the left rectangular approximation formula, we have

$$\int_t^{t+h} f(r, x(r))dr \approx hf(t, x(t))$$

which leads to the Euler’s method

$$x(t + h) = x(t) + hf(t, x(t))$$

On the other hand, the trapezoid rule leads to

$$\int_t^{t+h} f(r, x(r))dr \approx \frac{h}{2} [f(t, x(t)) + f(t + h, x(t + h))]$$

and the implicit formula,

$$x(t + h) = x(t) + \frac{h}{2} [f(t, x(t)) + f(t + h, x(t + h))]$$

because $x(t+h)$ appears on both sides
The Taylor series of a function
\[ x(t + h) = x(t) + hx'(t) + \frac{1}{2!}h^2 x''(t) + \frac{1}{3!}h^3 x'''(t) + \frac{1}{4!}h^4 x^{(4)}(t) + \ldots + \frac{1}{m!}h^m x^{(m)}(t) + \ldots \]
gives the numerical solution of \(x(t+h)\) when we truncate the series after the \((m+1)st\) terms.

If \(h\) is small, and if we know
\[ x(t), x'(t), x''(t), \ldots, x^{(m)}(t) \]
the Taylor series computes very accurate value of \(x(t+h)\).

If we truncate the series after the \((m+1)st\) term, the method is called the **Taylor series method of order \(m\)**

The Taylor series method of order 1 is known as the **Euler’s method**
Euler’s Method

The initial value problem
\[
\begin{align*}
x' &= f(t, x(t)) \\
x(a) &= x_a
\end{align*}
\]
over the interval \([a, b]\), we use the first two terms of the Taylor series
\[
x(t + h) \approx x(t) + hx'(t)
\]
This gives the formula of the Euler’s method
\[
x(t + h) = x(t) + hf(t, x(t))
\]
The computation starts from \(t = a\) and stops at \(t = b\) with \(n\) steps of size
\[
h = \frac{(b - a)}{n}
\]
Euler’s method

Using Euler’s method to solve an ODE. The typical behavior of the computed solution is that it runs away from the exact solution as more steps are taken, due to the accumulation of errors at each step.
Taylor Series Method of Higher Order

Consider the initial-value problem
\[ x' = 1 + x^2 + t^3 \]
\[ x(1) = -4 \]

We differentiate the equation several times, as
\[ x' = 1 + x^2 + t^3 \]
\[ x'' = 2xx' + 3t^2 \]
\[ x''' = 2xx'' + 2x'x' + 6t \]
\[ x^{(4)} = 2xx''' + 6x'x'' + 6 \]

These terms can be applied in order in the Taylor series method and the computed solution will be more accurate. For example, \( x(2) = 4.235841 \) from the Euler’s method. \( x(2) = 4.3712096 \) from the use of the 4\(^{th}\) derivative. The exact value to five significant figures is \( x(2) = 4.3712 \)
Type of Errors

When we truncate the Taylor series, we introduce the local truncation error

$$\frac{1}{120} h^5 x^5(\xi)$$

The truncation error is of order $h^5$, or $O(h^5)$.

The second type of error is the accumulation effects of all local truncation errors

The computed value of $x(t+h)$ is in error because $x(t)$ is already in error, due to the previous truncation error, and the current step involves another truncation error

The roundoff error can also contribute to the accumulation of error

These errors may be magnified by succeeding steps
Runge-Kutta Methods

The high order Taylor method for solving the initial-value problem

\[
\begin{align*}
    x' &= f(t, x) \\
    x(a) &= x_a
\end{align*}
\]

needs the derivatives of \( x \) by differentiating the function \( f \). This is not convenient. We need some highly accurate methods that make use of \( f \) directly.

The Runge-Kutta methods are a class of methods using multiple evaluations of \( f \), not its derivatives, to enhance computational accuracy.

We will illustrate the procedure to derive the Runge-Kutta method of order 2, and give the formula for Runge-Kutta method of order 4, which is popularly used.

We assume that \( f(t, x) \) can be computed for any \((t, x)\), so that \( x(t), x(t+h), x(t+2h), ... \), can be computed.
Taylor Series for $f(x,y)$

We can expand the Taylor series in two variables

$$f(x + h, y + k) = \sum_{i=0}^{\infty} \frac{1}{i!} \left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^i f(x, y)$$

The first few right-hand terms are

$$\left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^0 f(x, y) = f$$

$$\left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^1 f(x, y) = h \frac{\partial f}{\partial x} + k \frac{\partial f}{\partial y}$$

$$\left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^2 f(x, y) = h^2 \frac{\partial^2 f}{\partial x^2} + 2hk \frac{\partial^2 f}{\partial x\partial y} + k^2 \frac{\partial^2 f}{\partial y^2}$$

...
Taylor Series for \( f(x, y) \)

The Taylor series can be truncated as

\[
f(x + h, y + k) = \sum_{i=0}^{n-1} \frac{1}{i!} \left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^i f(x, y) + \frac{1}{n!} \left( h \frac{\partial}{\partial x} + k \frac{\partial}{\partial y} \right)^n f(x, y)
\]

We now use subscripts to denote the partial derivatives

\[
f(x + h, y + k) = f + (hf_x + kf_y)
\]

\[
+ \frac{1}{2!} \left( h^2 f_{xx} + 2hkf_{xy} + k^2 f_{yy} \right)
\]

\[
+ \frac{1}{3!} \left( h^3 f_{xxx} + 3h^2 kf_{xxy} + 3hk^2 f_{xyy} + k^3 f_{yyy} \right)
\]

\[+ \cdots
\]

\[
f(x + h, y) = f + hf_x + \frac{h^2}{2!} f_{xx} + \frac{h^3}{3!} f_{xxx} + \cdots
\]

\[
f(x, y + k) = f + kf_y + \frac{k^2}{2!} f_{yy} + \frac{k^3}{3!} f_{yyy} + \cdots
\]
Runge-Kutta Method of Order 2

Define two function evaluations

\[
\begin{cases}
K_1 = hf(t, x) \\
K_2 = hf(t + \alpha h, x + \beta K_1)
\end{cases}
\]

We add the linear combination of these quantities to the value of \( x \) at \( t \) to obtain the value at \( t+h \)

\[
x(t + h) = x(t) + w_1 K_1 + w_2 K_2
\]

or

\[
x(t + h) = x(t) + w_1 hf(t, x) + w_2 hf(t + \alpha h, x + \beta hf(t, x))
\]

We want to determine the constants in the above expression so that it will be as accurate as possible.

In other words, we want the above expression to match the Taylor series to as many terms as possible.
The Euler’s method (left) requires one function evaluation at each step. The second-order Runge-Kutta method (right) requires two function evaluations at each step.
Runge-Kutta Method of Order 2

Compare the intended formula
\[ x(t + h) = x(t) + w_1 hf(t, x) + w_2 hf(t + \alpha h, x + \beta hf(t, x)) \]
with the Taylor series
\[ x(t + h) = x(t) + hx'(t) + \frac{1}{2!} h^2 x''(t) + \frac{1}{3!} x'''(t) + \cdots \]

If we set \( w_1 = 1 \) and \( w_2 = 0 \), the two expressions agree to the \( h \) term.

The formula obtained is
\[ x(t + h) = x(t) + hf(t, x) \]
which is the Euler’s method, since
\[ x'(t) = f(t, x) \]
Runge-Kutta Method of Order 2

Revisit the two variable Taylor series

\[ f(t + \alpha h, x + \beta hf') = f + \alpha hf_t + \beta hff_x + \frac{1}{2} \left( ah \frac{\partial}{\partial t} + \beta hf \frac{\partial}{\partial x} \right)^2 f(x, y) \]

Using this formula in

\[ x(t + h) = x(t) + w_1 hf'(t, x) + w_2 hf'(t + \alpha h, x + \beta hf'(t, x)) \]

We have

\[ x(t + h) = x(t) + (w_1 + w_2)hf + \alpha w_2 h^2 f_t + \beta w_2 h^2 f f_x + O(h^3) \]

Note that

\[ x'' = \frac{dx'}{dt} = \frac{df(t, x)}{dt} = \left( \frac{\partial f}{\partial t} \right) \left( \frac{dt}{d} \right) + \left( \frac{\partial f}{\partial x} \right) \left( \frac{dx}{d} \right) = f_t + f_x f \]
Runge-Kutta Method of Order 2

The Taylor series equation

\[ x(t + h) = x(t) + hx'(t) + \frac{1}{2} h^2 x''(t) + \frac{1}{3!} h^3 x'''(t) + \cdots \]

becomes

\[ x(t + h) = x + hf + \frac{1}{2} h^2 f_t + \frac{1}{2} h^2 f_{ff} + O(h^3) \]

Compare the two expressions of \( x(t+h) \), we have

\[ w_1 + w_2 = 1 \quad \alpha w_2 = \frac{1}{2} \quad \beta w_2 = \frac{1}{2} \]

One solution is

\[ \alpha = 1 \quad \beta = 1 \quad w_1 = \frac{1}{2} \quad w_2 = \frac{1}{2} \]

Other solutions are possible, but the truncation errors will be the same
Runge-Kutta Method of Order 2

So the Runge-Kutta Method of Order 2 is

\[ x(t + h) = x(t) + \frac{h}{2} f(t, x) + \frac{h}{2} f(t + h, x + hf(t, x)) \]

Or equivalently

\[ x(t + h) = x(t) + \frac{1}{2} (K_1 + K_2) \]

where

\[ \begin{cases} 
K_1 = hf(t, x) \\
K_2 = hf(t + h, x + K_1) 
\end{cases} \]

The solution function at \( t+h \) is computed at the expenses of two evaluations of the function \( f \)

The truncation error of the second-order Runge-Kutta method is \( O(h^3) \)
The initial-value problem is

\[ x'(t) = -2x(t) + 3e^{-4t}, \quad x(0) = 1 \]
The most popular Runge-Kutta Method is that of order 4, as

\[ x(t + h) = x(t) + \frac{1}{6} \left( K_1 + K_2 + K_3 + K_4 \right) \]

where

\[
\begin{align*}
K_1 &= hf(t, x) \\
K_2 &= hf \left( t + \frac{1}{2} h, x + \frac{1}{2} K_1 \right) \\
K_3 &= hf \left( t + \frac{1}{2} h, x + \frac{1}{2} K_2 \right) \\
K_4 &= hf \left( t + h, x + K_3 \right)
\end{align*}
\]

This formula requires four function evaluations

The truncation error of the fourth-order Runge-Kutta method is \( O(h^5) \)
Fourth-Order Runge-Kutta Method

Illustration of Runge-Kutta method of order 4. Calculations are made at the initial time, two at half of the stepsize beyond the initial time and at the final time. These four calculations allow the use of larger overall stepsize with good accuracy.
Fourth-Order Runge-Kutta Method

The initial-value problem is \( x'(t) = y + x, \quad x(0) = 1 \)
Some Amazing Simulation Results
Adaptive Runge-Kutta Methods

In practical computations, we need to know the magnitude of errors involved in the computation.

Given a tolerance of error, we want to be assured that the computed numerical solution must not deviate from the true solution beyond the tolerance.

If a method is selected, the error tolerance dictates the allowable step size.

A uniform step size may not be desirable.

If the solution is smooth, a large step size may be used to reduce computational cost.

It is preferable that the method can automatically adjust the step size during the computation.
Compute the Errors

To advance the solution curve from $t$ to $t+h$, we can take one step of size $h$ using the fourth-order Runge-Kutta method.

We can also take two steps of size $h/2$ to arrive at $t+h$.

If there were no truncation errors, the values of the numerical solution $x(t+h)$ computed from both procedures would be the same.

The difference in the numerical results can be taken as an estimate of the local truncation error.

If this error is within the prescribed tolerance, the current step size $h$ is satisfactory. If the difference exceeds the tolerance, the step size is halved.

This procedure is easily programmed but rather wasteful of computing time.
**Fehlberg Method of Order 4**

The formula is

\[ x(t + h) = x(t) + \frac{25}{216} K_1 + \frac{1408}{2565} K_3 + \frac{2197}{4104} K_4 - \frac{1}{5} K_5 \]

where

\[
\begin{align*}
K_1 &= hf(t, x) \\
K_2 &= hf\left(t + \frac{1}{4} h, x + \frac{1}{4} K_1\right) \\
K_3 &= hf\left(t + \frac{3}{8} h, x + \frac{3}{32} K_1 + \frac{9}{32} K_2\right) \\
K_4 &= hf\left(t + \frac{12}{13} h, x + \frac{1932}{2197} K_1 - \frac{7200}{2197} K_2 + \frac{7296}{2197} K_3\right) \\
K_5 &= hf\left(t + h, x + \frac{439}{216} K_1 - 8K_2 + \frac{3680}{513} K_3 - \frac{845}{4104} K_4\right)
\end{align*}
\]
The Fehlberg method of order 4 requires one more function evaluation than the classical Runge-Kutta method, it is not very attractive.

However, with one more function evaluation, as

\[ K_6 = hf \left( t + \frac{1}{2} h, x - \frac{8}{27} K_1 + 2K_2 - \frac{3544}{12825} K_3 + \frac{1859}{4104} K_4 - \frac{11}{40} K_5 \right) \]

We can obtain the fifth-order Runge-Kutta method, i.e.,

\[ x(t + h) = x(t) + \frac{16}{135} K_1 + \frac{6656}{12825} K_3 + \frac{28561}{56430} K_4 - \frac{9}{50} K_5 + \frac{2}{55} K_6 \]

The difference between the values of \( x(t+h) \) computed from the fourth and fifth order procedures is an estimate of the local truncation error in the fourth-order procedure. So six function evaluations give a fourth-order approximation, together with an error estimate.
Adaptive Process

1. Given a step size $h$ and an initial value $x(t)$, the RK45 routine computes the value $x(t+h)$ and an error estimate $\varepsilon$

2. If $\varepsilon_{\text{min}} \leq \varepsilon \leq \varepsilon_{\text{max}}$, then the step size $h$ is not changed and the next step size is taken by replacing step 1 with initial value $x(t+h)$

3. If $\varepsilon \leq \varepsilon_{\text{min}}$, then $h$ is replaced by $2h$, provided $|2h| \leq h_{\text{max}}$

4. If $\varepsilon > \varepsilon_{\text{max}}$, then $h$ is replaced by $h/2$, provided $|h/2| \geq h_{\text{min}}$

5. If $h_{\text{min}} \leq |h| \leq h_{\text{max}}$, then the step size is repeated by returning to step 1 with $x(t)$ and the new $h$ value

Here $\varepsilon_{\text{max}}, \varepsilon_{\text{min}}$ are the maximum and minimum error tolerances allowed

$h_{\text{max}}, h_{\text{min}}$ are the maximum and minimum step sizes allowed
Adams-Bashforth-Moulton Formulas

Let us consider the first-order ordinary differential equation

\[ x'(t) = f(t, x(t)) \]

If we assume the values of the unknown function have been computed at several points to the left of \( t \), i.e., \( t, t-h, t-2h, \ldots, t-(n-1)h \), we want to compute \( x(t+h) \). Using the theorem of calculus, we have

\[
x(t + h) = x(t) + \int_t^{t+h} x'(s) \, ds
= x(t) + \int_t^{t+h} f(s, x(s)) \, ds
\approx x(t) + \sum_{j=1}^{n} c_j f_j
\]

where the abbreviation \( f_j \) is

\[ f_j = f(t - (j-1)h, x(t-(j-1)h)) \]
Adams-Bashforth Formulas

The last step needs a rule of the form

$$\int_0^1 F(r)dr \approx c_1 F(0) + c_2 F(-1) + \cdots + c_n F(1-n)$$

We need to determine the $n$ coefficients $c_j$. We can insist on integrating each function $1, r, r^2, \cdots, r^{n-1}$ exactly. The appropriate equation is

$$\int_0^1 r^{i-1}dr = \sum_{j=1}^{n} c_j (1-j)^{i-1} \quad (1 \leq i \leq n)$$

A linear system of the form $Ax=b$ of $n$ equations can be solved for $n$ unknowns. The coefficients of the matrix $A$ are

$$A_{ij} = (1-j)^{i-1}$$

and the right-hand side is $b_i = 1/i$
Adams-Moulton Formulas

In Adams-Moulton formulas, the quadrature is of the form

\[ \int_{0}^{1} G(r) dr \approx \sum_{j=1}^{n} C_j G(2 - j) \]

Use the change of variable from \( s \) to \( p \) given by \( s = hp - t \), the new integral will be

\[ h \int_{0}^{1} g( hp + t ) dp \]

We now have

\[ \int_{t}^{t+h} F(r) dr \approx \frac{h}{24} \left[ 55 F(t) - 59 F(t - h) + 37 F(t - 2h) - 9 F(t - 3h) \right] \]

\[ \int_{t}^{t+h} G(r) dr \approx \frac{h}{24} \left[ 9G(t + h) + 19G(t) - 5G(t - h) - 249G(t - 2h) \right] \]
Stability Analysis

For some initial value problems, a small error in the initial value may produce errors of large magnitude, and the computed solution is completely wrong.

The truncation error in each step further deteriorates the computed solution.

For other initial value problems, the situation is not that severe.
A Case of Diverging Solutions

The initial values are close to each other. But over the time, the solutions diverge to different paths
A Case of Converging Solutions

The initial values are quite different. But over the time, the solutions converge to something very close to each other.
For the general differential equation
\[
\begin{align*}
    x' &= f(t, x) \\
    x(a) &= s
\end{align*}
\]

If
\[f_x > \delta \text{ for some } \delta > 0\]
the curves diverge. If
\[f_x < -\delta \text{ for some } \delta > 0\]
then the curves converge.