TMV035 Analysis and Linear Algebra B, 2002

ORDINARY DIFFERENTIAL EQUATIONS—SUMMARY

In this lecture we present a summary of all our work on ordinary differential equations, from the integral and the fundamental theorem of calculus, via the exponential function, the trigonometric and hyperbolic functions, to the general initial value problem for systems of ODEs.

1.1. **The general initial-value problem.** The general system of n equations in n unknown functions:

$$\begin{aligned} u'(x) &= f(x, u(x)), \quad x \in [a, b] \\ u(a) &= u_a \end{aligned} \tag{1}$$

The solution is constructed by the *algorithm*:

$$\begin{aligned} &x_0 = a,\ U(x_0) = u_a \\ &x_i = x_{i-1} + h \\ &U(x_i) = U(x_{i-1}) + hf(x_{i-1}, U(x_{i-1})) \end{aligned} \tag{2}$$

Motivation: if u satisfies (1), then

$$u(x_i) \approx u(x_{i-1}) + hu'(x_{i-1}) = u(x_{i-1}) + hf(x_{i-1}, u(x_{i-1})) \tag{3}$$

Matlab command:

```
>> [x,U]=my_ode(f,I,ua,h);  plot(x,U)
```

In the following sections we repeat the *special differential equations* in order to see how they fit into the form of the *general system*. The special differential equations are those for which we can express the solution analytically in terms of the *special functions* log, exp, sin, cos, etc., i.e., the equations for which we have a *solution formula*.

1.2. **The integral.**

$$\begin{aligned} u'(x) &= f(x), \quad x \in [a, b] \\ u(a) &= u_a \end{aligned} \tag{4}$$

The solution can be expressed as an *integral*:

$$u(x) = u_a + \int_a^x f(y)\,\mathrm{d}y \tag{5}$$

It is constructed by the algorithm:

$$\begin{aligned} &x_0 = a,\ U(x_0) = u_a \\ &x_i = x_{i-1} + h \\ &U(x_i) = U(x_{i-1}) + hf(x_{i-1}) \end{aligned} \tag{6}$$

Example 1.

$$\begin{aligned} u'(x) &= x^3, \quad x \in [2, 5] \\ u(2) &= 3 \end{aligned} \tag{7}$$

The solution formula is:

$$u(x) = 3 + \int_2^x y^3\,\mathrm{d}y = 3 + \Big[\frac{y^4}{4}\Big]_2^x = 3 + \frac{x^4 - 16}{4} \tag{8}$$

Matlab function file:

```
function y=funk1(x,u)
y=x^3;
```

Matlab command:

Date: November 17, 2002, Stig Larsson, Computational Mathematics, Chalmers University of Technology.

```
>> [x,U]=my_ode('funk1',[2 5],3,1e-2);  plot(x,U)
```

□

Example 2.

$$\begin{aligned} u'(x) &= 1/x, \quad x \in [1,5] \\ u(1) &= 0 \end{aligned} \tag{9}$$

The solution is called the *natural logarithm*:

$$u(x) = \int_1^x \frac{1}{y}\,\mathrm{d}y = \log(x) \tag{10}$$

MATLAB function file:

```
function y=funk2(x,u)
y=1/x;
```

MATLAB command:

```
>> [x,U]=my_ode('funk2',[1 5],0,1e-2);  plot(x,U)
```

□

1.3. **The exponential function.**

$$\begin{aligned} u'(x) &= u(x), \quad x \in [0,b] \\ u(0) &= 1 \end{aligned} \tag{11}$$

The solution is called the *exponential function*:

$$u(x) = \exp(x) \tag{12}$$

It is constructed by the algorithm:

$$\begin{aligned} &x_0 = 0,\ U(x_0) = 1 \\ &x_i = x_{i-1} + h \\ &U(x_i) = U(x_{i-1}) + hU(x_{i-1}) \end{aligned} \tag{13}$$

More generally:

$$\begin{aligned} u'(x) &= cu(x), \quad x \in [a,b] \\ u(a) &= u_a \end{aligned} \tag{14}$$

The solution can be expressed by means of the exponential function:

$$u(x) = u_a \exp(c(x-a)) \tag{15}$$

It is constructed by the algorithm:

$$\begin{aligned} &x_0 = a,\ U(x_0) = u_a \\ &x_i = x_{i-1} + h \\ &U(x_i) = U(x_{i-1}) + hcU(x_{i-1}) \end{aligned} \tag{16}$$

Example 3.

$$\begin{aligned} u'(x) &= -3u(x), \quad x \in [0,5] \\ u(0) &= 4 \end{aligned} \tag{17}$$

The solution formula is:

$$u(x) = 4\exp(-3x) \tag{18}$$

MATLAB function file:

```
function y=funk3(x,u)
y=-3*u;
```

MATLAB command:

```
>> [x,U]=my_ode('funk3',[0 5],4,1e-2);  plot(x,U)
```

□

Example 4.

$$u'(x) = -3u(x), \quad x \in [2,5] \\ u(2) = 4 \tag{19}$$

The solution formula is:

$$u(x) = 4\exp(-3(x-2)) \tag{20}$$

Matlab function file:

```
function y=funk3(x,u)
y=-3*u;
```

Matlab command:

```
>> [x,U]=my_ode('funk3',[2 5],4,1e-2);  plot(x,U)
```

□

Even more generally:

$$u'(x) = c(x)u(x) + f(x), \quad x \in [a,b] \\ u(a) = u_a \tag{21}$$

The solution can be expressed by means of the exponential function AMBS Chapter 35:

$$u(x) = u_a \exp(C(x)) + \int_a^x \exp(C(x) - C(y))f(y)\,\mathrm{d}y, \quad C(x) = \int_a^x c(y)\,\mathrm{d}y \tag{22}$$

The formula is obtained by multiplying the differential equation by the *integrating factor* $\exp(-C(x))$ and then integrating the resulting equation.

The solution is constructed by the algorithm:

$$\begin{aligned} &x_0 = a,\ U(x_0) = u_a \\ &x_i = x_{i-1} + h \\ &U(x_i) = U(x_{i-1}) + h\big(c(x_{i-1})U(x_{i-1}) + f(x_{i-1})\big) \end{aligned} \tag{23}$$

Example 5.

$$u'(x) = -3u(x) + 5, \quad x \in [2,5] \\ u(2) = 4 \tag{24}$$

The solution formula is, with $C(x) = \int_2^x (-3)\,\mathrm{d}y = -3(x-2)$:

$$u(x) = 4\exp(-3(x-2)) + \frac{5}{3}(1 - \exp(-3(x-2))) \tag{25}$$

Matlab function file:

```
function y=funk4(x,u)
y=-3*u+5;
```

Matlab command:

```
>> [x,U]=my_ode('funk4',[2 5],4,1e-2);  plot(x,U)
```

□

Example 6.

$$u'(x) = -xu(x), \quad x \in [0,2] \\ u(0) = 4 \tag{26}$$

The solution formula is, with $C(x) = \int_0^x (-y)\,\mathrm{d}y = -x^2/2$:

$$u(x) = 4\exp(-x^2/2) \tag{27}$$

Matlab function file:

```
function y=funk5(x,u)
y=-x*u;
```

Matlab command:

```
>> [x,U]=my_ode('funk5',[0 2],4,1e-2);  plot(x,U)
```

□

1.4. **The trigonometric functions.**

$$\begin{aligned} &u''(x)+u(x)=0, \quad x\in[0,b] \\ &u(0)=0,\ u'(0)=1 \end{aligned} \tag{28}$$

This is an equation of second order. We re-write it as a system of two equations of first order:

$$w=\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}=\begin{bmatrix} u \\ u' \end{bmatrix}, \quad w_0=\begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad w'=\begin{bmatrix} w_1' \\ w_2' \end{bmatrix}=\begin{bmatrix} u' \\ u'' \end{bmatrix}=\begin{bmatrix} u' \\ -u \end{bmatrix}=\begin{bmatrix} w_2 \\ -w_1 \end{bmatrix}=\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \tag{29}$$

The equation becomes

$$\begin{aligned} &w'(x)=Aw(x), \quad x\in[0,b] \\ &w(0)=w_0 \end{aligned} \qquad w_0=\begin{bmatrix} 0 \\ 1 \end{bmatrix},\ A=\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{30}$$

The solutions are called *trigonometric functions*:

$$w(x)=\begin{bmatrix} w_1(x) \\ w_2(x) \end{bmatrix}=\begin{bmatrix} u(x) \\ u'(x) \end{bmatrix}=\begin{bmatrix} \sin(x) \\ \cos(x) \end{bmatrix} \tag{31}$$

They are constructed by the algorithm:

$$\begin{aligned} &x_0=0,\ W(x_0)=w_0 \\ &x_i=x_{i-1}+h \\ &W(x_i)=W(x_{i-1})+hAW(x_{i-1}) \end{aligned} \qquad w_0=\begin{bmatrix} 0 \\ 1 \end{bmatrix},\ A=\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{32}$$

More generally:

$$\begin{aligned} &u''(x)+c^2u(x)=0, \quad x\in[a,b] \\ &u(a)=u_0,\ u'(a)=u_1 \end{aligned} \tag{33}$$

The solution can be expressed by means of the trigonometric functions:

$$u(x)=u_0\cos(c(x-a))+\frac{u_1}{c}\sin(c(x-a)) \tag{34}$$

It is constructed by the algorithm:

$$\begin{aligned} &x_0=a,\ W(x_0)=w_0 \\ &x_i=x_{i-1}+h \\ &W(x_i)=W(x_{i-1})+hAW(x_{i-1}) \end{aligned} \qquad w_0=\begin{bmatrix} u_0 \\ u_1 \end{bmatrix},\ A=\begin{bmatrix} 0 & 1 \\ -c^2 & 0 \end{bmatrix} \tag{35}$$

Example 7.

$$\begin{aligned} &u''(x)+4u(x)=0, \quad x\in[0,5] \\ &u(0)=3,\ u'(0)=1 \end{aligned} \tag{36}$$

The solution formula is:

$$u(x)=3\cos(2x)+\frac{1}{2}\sin(2x) \tag{37}$$

Matlab function file:

```
function y=funk6(x,w)
A=[0 1;-4 0];
y=A*w;
```

Matlab command:

```
>> [x,W]=my_ode('funk6',[0 5],[3;1],1e-2);  plot(x,W), plot(W(:,1),W(:,2))
```

□

Example 8.

$$u''(x) + 4u(x) = 0, \quad x \in [2,5] \\ u(2) = 3,\ u'(2) = 0 \tag{38}$$

The solution formula is:

$$u(x) = 3\cos(2(x-2)) \tag{39}$$

MATLAB function file:

```
function y=funk6(x,w)
A=[0 1;-4 0];
y=A*w;
```

MATLAB command:

```
>> [x,W]=my_ode('funk6',[2 5],[3;0],1e-2);  plot(x,W), plot(W(:,1),W(:,2))
```

□

1.5. The hyperbolic functions.

$$u''(x) - u(x) = 0, \quad x \in [0,b] \\ u(0) = 0,\ u'(0) = 1 \tag{40}$$

This is an equation of second order. We re-write it as a system of two equations of first order. The equation becomes

$$\begin{aligned} w'(x) &= Aw(x), \quad x \in [0,b] \\ w(0) &= w_0 \end{aligned} \qquad w_0 = \begin{bmatrix} 0 \\ 1 \end{bmatrix},\ A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{41}$$

The solutions are called *hyperbolic functions*:

$$w(x) = \begin{bmatrix} w_1(x) \\ w_2(x) \end{bmatrix} = \begin{bmatrix} u(x) \\ u'(x) \end{bmatrix} = \begin{bmatrix} \sinh(x) \\ \cosh(x) \end{bmatrix} \tag{42}$$

They are constructed by the algorithm:

$$\begin{aligned} &x_0 = 0,\ W(x_0) = w_0 \\ &x_i = x_{i-1} + h \\ &W(x_i) = W(x_{i-1}) + hAW(x_{i-1}) \end{aligned} \qquad w_0 = \begin{bmatrix} 0 \\ 1 \end{bmatrix},\ A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{43}$$

They can also be expressed by means of the exponential function:

$$\sinh(x) = \frac{e^x - e^{-x}}{2}, \quad \cosh(x) = \frac{e^x + e^{-x}}{2} \tag{44}$$

see AMBS Chapter 32.5.

More generally:

$$u''(x) - c^2u(x) = 0, \quad x \in [a,b] \\ u(a) = u_0,\ u'(a) = u_1 \tag{45}$$

The solution can be expressed by means of the hyperbolic functions:

$$u(x) = u_0\cosh(c(x-a)) + \frac{u_1}{c}\sinh(c(x-a)) \tag{46}$$

It is constructed by the algorithm:

$$\begin{aligned} &x_0 = a,\ W(x_0) = w_0 \\ &x_i = x_{i-1} + h \\ &W(x_i) = W(x_{i-1}) + hAW(x_{i-1}) \end{aligned} \qquad w_0 = \begin{bmatrix} u_0 \\ u_1 \end{bmatrix},\ A = \begin{bmatrix} 0 & 1 \\ c^2 & 0 \end{bmatrix} \tag{47}$$

Example 9.

$$u''(x) - u(x) = 0, \quad x \in [0,3]$$
$$u(0) = 1,\ u'(0) = 0 \tag{48}$$

The solution formula is:

$$u(x) = \cosh(x) = \frac{1}{2}(e^x + e^{-x}) \tag{49}$$

Matlab function file:

```
function y=funk7(x,w)
A=[0 1;1 0];
y=A*w;
```

Matlab command:

```
>> [x,W]=my_ode('funk7',[0 3],[1;0],1e-2);  plot(x,W), plot(W(:,1),W(:,2))
```

□

Example 10.

$$u''(x) - 4u(x) = 0, \quad x \in [2,5]$$
$$u(2) = 3,\ u'(2) = 0 \tag{50}$$

The solution formula is:

$$u(x) = 3\cosh(2(x-2)) = \frac{3}{2}(e^{2(x-2)} + e^{-2(x-2)}) \tag{51}$$

Matlab function file:

```
function y=funk8(x,w)
A=[0 1;4 0];
y=A*w;
```

Matlab command:

```
>> [x,W]=my_ode('funk8',[2 5],[3;0],1e-2);  plot(x,W), plot(W(:,1),W(:,2))
```

□

1.6. **The general system, again.**

Example 11. The Volterra-Lotka equations AMBS Chapter 39.3:

$$u_1'(t) = au_1(t) - bu_1(t)u_2(t)$$
$$u_2'(t) = -cu_2(t) + du_1(t)u_2(t) \tag{52}$$

Matlab function file:

```
function y=volterra(t,u)
a=.5; b=1; c=.2; d=1;
y=zeros(2,1);
y(1)= a*u(1)-b*u(1)*u(2);
y(2)=-c*u(2)+d*u(1)*u(2);
```

Matlab command:

```
>> [t,U]=my_ode('volterra',[0 50],[.5;.3],1e-2);  plot(t,U), plot(U(:,1),U(:,2))
```

□