UiO : Department of Mathematics
University of Oslo

Life Insurance and Finance

Lecture 2: Numerical methods for ODEs

David R. Banos

STK4500

1 ODEs

2 Euler's method

ODEs

Let $d \geq 0$ and $(t_0, x_0) \in \mathbb{R} \times \mathbb{R}^d$ some fixed starting time and position.
Let $f : [t_0, \infty) \times \mathbb{R}^d \to \mathbb{R}^d$ be a time dependent vector field.
A first-order differential equation is a Cauchy problem, also called initial value problem (IVP) of the form

$$x'(t) = f(t, x(t)), \quad t \in [t_0, \infty), \quad x(t_0) = x_0 \in \mathbb{R}^d. \tag{1}$$

Let $d \geq 0$ and $(t_0, x_0) \in \mathbb{R} \times \mathbb{R}^d$ some fixed starting time and position. Let $f : [t_0, \infty) \times \mathbb{R}^d \to \mathbb{R}^d$ be a time dependent vector field.
A first-order differential equation is a Cauchy problem, also called initial value problem (IVP) of the form

$$x'(t) = f(t, x(t)), \quad t \in [t_0, \infty), \quad x(t_0) = x_0 \in \mathbb{R}^d. \tag{1}$$

Example

If $f(t, y) = y$ then $f(t, x(t)) = x(t)$ and hence $x'(t) = x(t)$.

Let $d \geq 0$ and $(t_0, x_0) \in \mathbb{R} \times \mathbb{R}^d$ some fixed starting time and position. Let $f : [t_0, \infty) \times \mathbb{R}^d \to \mathbb{R}^d$ be a time dependent vector field.
A first-order differential equation is a Cauchy problem, also called initial value problem (IVP) of the form

$$x'(t) = f(t, x(t)), \quad t \in [t_0, \infty), \quad x(t_0) = x_0 \in \mathbb{R}^d. \tag{1}$$

Example

If $f(t, y) = y$ then $f(t, x(t)) = x(t)$ and hence $x'(t) = x(t)$. A function who derivative is itself is of the form $x(t) = Ce^t$. If $x(t_0) = x_0$ then $x(t) = x_0 e^{t-t_0}$, $t \in [t_0, \infty)$.

First-order means that only the first derivative of x appears in the equation, and higher derivatives are absent.

Higher-order differential equations can be reduced to first-order by increasing the dimension.

First-order means that only the first derivative of x appears in the equation, and higher derivatives are absent.

Higher-order differential equations can be reduced to first-order by increasing the dimension.

For example, the second-order equation $x''(t) = f(t, x(t))$ can be reduced by defining $y(t) \triangleq x'(t)$.

Then $y'(t) = x''(t) = f(t, x(t))$ and we have the equation

$$z'(t) = F(t, z(t)),$$

where $z(t) = (x(t), y(t))^t$ and $F(t, u_1, u_2) = (u_2, f(t, u_1))^t$. We managed to do so by increasing the dimension by one.

Euler's method

We know that

$$x'(t) = \lim_{h\to 0} \frac{x(t+h) - x(t)}{h}.$$

We know that

$$x'(t) = \lim_{h\to 0} \frac{x(t+h)-x(t)}{h}.$$

Based on

$$x'(t) \approx \frac{x(t+h)-x(t)}{h}$$

for a small $h > 0$.

We know that

$$x'(t) = \lim_{h \to 0} \frac{x(t+h) - x(t)}{h}.$$

Based on

$$x'(t) \approx \frac{x(t+h) - x(t)}{h}$$

for a small $h > 0$. Recall

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

We know that

$$x'(t) = \lim_{h\to 0} \frac{x(t+h) - x(t)}{h}.$$

Based on

$$x'(t) \approx \frac{x(t+h) - x(t)}{h}$$

for a small $h > 0$. Recall

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

Subtitute:

$$\frac{x(t+h) - x(t)}{h} \approx f(t, x(t)).$$

We know that

$$x'(t) = \lim_{h \to 0} \frac{x(t+h) - x(t)}{h}.$$

Based on

$$x'(t) \approx \frac{x(t+h) - x(t)}{h}$$

for a small $h > 0$. Recall

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

Subtitute:

$$\frac{x(t+h) - x(t)}{h} \approx f(t, x(t)).$$

Isolate $x(t+h)$:

$$x(t+h) \approx x(t) + hf(t, x(t)).$$

We know that

$$x'(t) = \lim_{h \to 0} \frac{x(t+h) - x(t)}{h}.$$

Based on

$$x'(t) \approx \frac{x(t+h) - x(t)}{h}$$

for a small $h > 0$. Recall

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

Subtitute:

$$\frac{x(t+h) - x(t)}{h} \approx f(t, x(t)).$$

Isolate $x(t+h)$:

$$x(t+h) \approx x(t) + hf(t, x(t)).$$

If t is such that $x(t)$ is known, then we can guess x at a later time $t+h$

At t_0, $x(t_0) = x_0$ is known. Thus the value of x at time $t_0 + h$ can be approximated by

$$x(t_0 + h) \approx x(t_0) + hf(t, x(t_0)) = x_0 + hf(t, x_0).$$

At t_0, $x(t_0) = x_0$ is known. Thus the value of x at time $t_0 + h$ can be approximated by

$$x(t_0 + h) \approx x(t_0) + hf(t, x(t_0)) = x_0 + hf(t, x_0).$$

More generally, consider a partition of $[t_0, \infty)$ with points defined by

$$t_i \triangleq t_0 + ih, \quad i = 0, 1, \ldots$$

and denote

$$x_i \triangleq x(t_i)$$

the value of the solution at $t_i = t_0 + ih$.

At t_0, $x(t_0) = x_0$ is known. Thus the value of x at time $t_0 + h$ can be approximated by

$$x(t_0 + h) \approx x(t_0) + hf(t, x(t_0)) = x_0 + hf(t, x_0).$$

More generally, consider a partition of $[t_0, \infty)$ with points defined by

$$t_i \triangleq t_0 + ih, \quad i = 0, 1, \dots$$

and denote

$$x_i \triangleq x(t_i)$$

the value of the solution at $t_i = t_0 + ih$. Then, knowing x_i, that is (t_i, x_i), allows us to find x_{i+1} by

$$x_{i+1} = x_i + hf(t_i, x_i), \quad i = 0, 1, \dots$$

Example

$$x'(t) = x(t), \quad x(0) = 1.$$

Example

$$x'(t) = x(t), \quad x(0) = 1.$$

- Partition of points: $t_i = ih$, $h > 0$ step size, $i = 0, \ldots, n$.

Example

$$x'(t) = x(t), \quad x(0) = 1.$$

- Partition of points: $t_i = ih$, $h > 0$ step size, $i = 0, \ldots, n$.
- x_i is an approximation of x at t_i, i.e. $x_i \approx x(t_i)$.

Example

$$x'(t) = x(t), \quad x(0) = 1.$$

- Partition of points: $t_i = ih$, $h > 0$ step size, $i = 0, \ldots, n$.
- x_i is an approximation of x at t_i, i.e. $x_i \approx x(t_i)$.
- The first value is known, namely $x(t_0) = x(0) = x_0 = 1$. Then

$$x_{i+1} = x_i + hf(t, x_i) = x_i + hx_i = x_i(1 + h), \quad i = 0, \ldots, n.$$

Example

$$x'(t) = x(t), \quad x(0) = 1.$$

- Partition of points: $t_i = ih$, $h > 0$ step size, $i = 0, \ldots, n$.
- x_i is an approximation of x at t_i, i.e. $x_i \approx x(t_i)$.
- The first value is known, namely $x(t_0) = x(0) = x_0 = 1$. Then

$$x_{i+1} = x_i + hf(t, x_i) = x_i + hx_i = x_i(1 + h), \quad i = 0, \ldots, n.$$

- Recursively,

$$x_i = (1 + h)^i, \quad i = 0, \ldots, n.$$

Example (continued)

We know $x(t) = e^t$. What is the error we commit?

Example (continued)

We know $x(t) = e^t$. What is the error we commit?
The Global Truncation Error (GTE) we commit is:

$$\begin{aligned} GTE &\triangleq \max_{i=0,\dots,n} |x(t_i) - x_i| \\ &= \max_{i=0,\dots,n} |e^{ih} - (1+h)^i| \\ &\leq |e^{nh} - (1+h)^n| \\ &= \left| e - \left(1 + \frac{1}{n}\right)^n \right|. \end{aligned}$$

Example (continued)

We know $x(t) = e^t$. What is the error we commit?
The Global Truncation Error (GTE) we commit is:

$$\begin{aligned} GTE &\triangleq \max_{i=0,\dots,n} |x(t_i) - x_i| \\ &= \max_{i=0,\dots,n} |e^{ih} - (1+h)^i| \\ &\leq |e^{nh} - (1+h)^n| \\ &= \left| e - \left(1 + \frac{1}{n}\right)^n \right|. \end{aligned}$$

which goes to zero as $h \to 0$ or as $n \to \infty$.

Example (Disability model)

We wish to solve

$$\frac{d}{dt}P(s,t) = P(s,t)\Lambda(t)$$

if we go for the forward equation.

Example (Disability model)

We wish to solve

$$\frac{d}{dt}P(s,t) = P(s,t)\Lambda(t)$$

if we go for the forward equation.
The matrix P of the unknowns (unknown functions) is given by

$$P(s,t) = \begin{pmatrix} p_{**}(s,t) & p_{*\diamond}(s,t) & p_{*\dagger}(s,t) \\ p_{\diamond *}(s,t) & p_{\diamond\diamond}(s,t) & p_{\diamond\dagger}(s,t) \\ p_{\dagger *}(s,t) & p_{\dagger\diamond}(s,t) & p_{\dagger\dagger}(s,t) \end{pmatrix},$$

Example (Disability model)

We wish to solve

$$\frac{d}{dt}P(s,t) = P(s,t)\Lambda(t)$$

if we go for the forward equation.
The matrix P of the unknowns (unknown functions) is given by

$$P(s,t) = \begin{pmatrix} p_{**}(s,t) & p_{*\diamond}(s,t) & p_{*\dagger}(s,t) \\ p_{\diamond *}(s,t) & p_{\diamond\diamond}(s,t) & p_{\diamond\dagger}(s,t) \\ p_{\dagger *}(s,t) & p_{\dagger\diamond}(s,t) & p_{\dagger\dagger}(s,t) \end{pmatrix},$$

but the last row is exactly 0 0 1 which can be omitted. Hence, we rather look at

$$P(s,t) = \begin{pmatrix} p_{**}(s,t) & p_{*\diamond}(s,t) \\ p_{\diamond *}(s,t) & p_{\diamond\diamond}(s,t) \end{pmatrix}.$$

Example (continued)

The (matrix) vector field in this case is a linear 2×2-transformation

$$f(t, M) = M \cdot \Lambda(t),$$

where $t \geq 0$ and M is a 2×2-matrix. **NB!** Respect the order of the matrices.

Example (continued)

The (matrix) vector field in this case is a linear 2×2-transformation

$$f(t, M) = M \cdot \Lambda(t),$$

where $t \geq 0$ and M is a 2×2-matrix. **NB!** Respect the order of the matrices.

$$\Lambda(t) = \begin{pmatrix} \mu_{**}(t) & \mu_{*\diamond}(t) \\ \mu_{\diamond *}(t) & \mu_{\diamond\diamond}(t) \end{pmatrix}.$$

Hence,

$$\frac{d}{dt} P(s,t) = P(s,t)\Lambda(t), \quad t \geq 0, \quad P(s,s) = Id.$$

Example (continued)

The (matrix) vector field in this case is a linear 2×2-transformation

$$f(t, M) = M \cdot \Lambda(t),$$

where $t \geq 0$ and M is a 2×2-matrix. **NB!** Respect the order of the matrices.

$$\Lambda(t) = \begin{pmatrix} \mu_{**}(t) & \mu_{*\diamond}(t) \\ \mu_{\diamond *}(t) & \mu_{\diamond\diamond}(t) \end{pmatrix}.$$

Hence,

$$\frac{d}{dt}P(s,t) = P(s,t)\Lambda(t), \quad t \geq 0, \quad P(s,s) = Id.$$

Next step: discretize time and approximate $P(s, t_i)$, $i = 0, dots$.

Example (continued)

Take small h. Let $t_i \triangleq s + ih$, $i \geq 0$.

Example (continued)

Take small h. Let $t_i \triangleq s + ih$, $i \geq 0$.
Denote by P_i an approximation of the matrix $P(s, t_i)$.

Example (continued)

Take small h. Let $t_i \triangleq s + ih$, $i \geq 0$.
Denote by P_i an approximation of the matrix $P(s, t_i)$.
Euler's method gives the following scheme:

$$P_{i+1} = P_i + hP_i\Lambda(t_i) = P_i\left(Id + h\Lambda(t_i)\right) = P_i\left(Id + h\Lambda(s + ih)\right), \quad i \geq 0.$$

Example (continued)

Take small h. Let $t_i \triangleq s + ih$, $i \geq 0$.
Denote by P_i an approximation of the matrix $P(s, t_i)$.
Euler's method gives the following scheme:

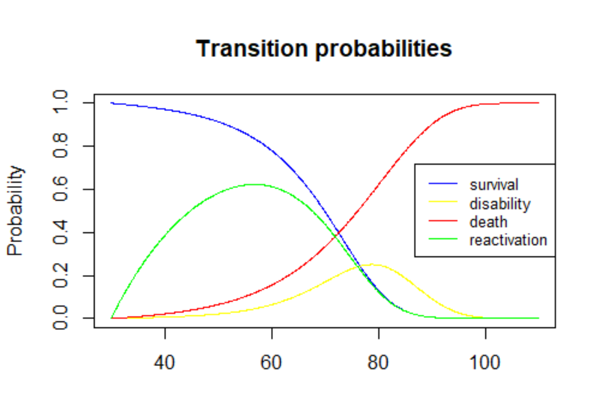
$$P_{i+1} = P_i + hP_i\Lambda(t_i) = P_i\left(Id + h\Lambda(t_i)\right) = P_i\left(Id + h\Lambda(s + ih)\right), \quad i \geq 0.$$

Let us use the matrix $\Lambda(t)$ from the book, see Example 2.4.2 on page 18 and Example 4.2.2 on page 30.

$$\mu_*(t) = \mu_{*\diamond}(t) + \mu_{*\dagger}(t), \quad \mu_{*\diamond}(t) = 0.0004 + 10^{0.06t-5.46},$$
$$\mu_{\diamond *}(t) = 0.05, \quad \mu_\diamond(t) = \mu_{\diamond *}(t) + \mu_{\diamond\dagger}(t).$$

Example (continued)

We start at age $s = 30$ and look at $t_i = 30 + ih$ with $h = \frac{1}{12}$ monthly steps. We run the algorithm until $t = 110$ years, i.e. $i = 0, 1, \ldots, 1080 = n$ where $n = 120 \cdot \frac{1}{h}$.



UiO : **Department of Mathematics**
University of Oslo

David R. Banos

Life Insurance and Finance

Lecture 2: Numerical methods for ODEs