Kom
verder

SAXION

Mathematical Modelling Numerical methods for differential equations in Matlab

Ronald Tangelder, Jan de Wilde

Kom verder

SAXION

Goals

- You have an overview of the simplest, but at the same time, the most important numerical solving methods.
- You have a number of typical terms which deal with numerical methods (*convergence order*, *error*, *step size* etc ...)
- You have understood that always a compromise between speed and precision of a numerical method has to be made.
- You have understood how to approach an investigation of the convergence of a numerical method.

Why numerical?

It is good to study and to understand analytical. In doing so you learn a lot about DEs and their solutions.

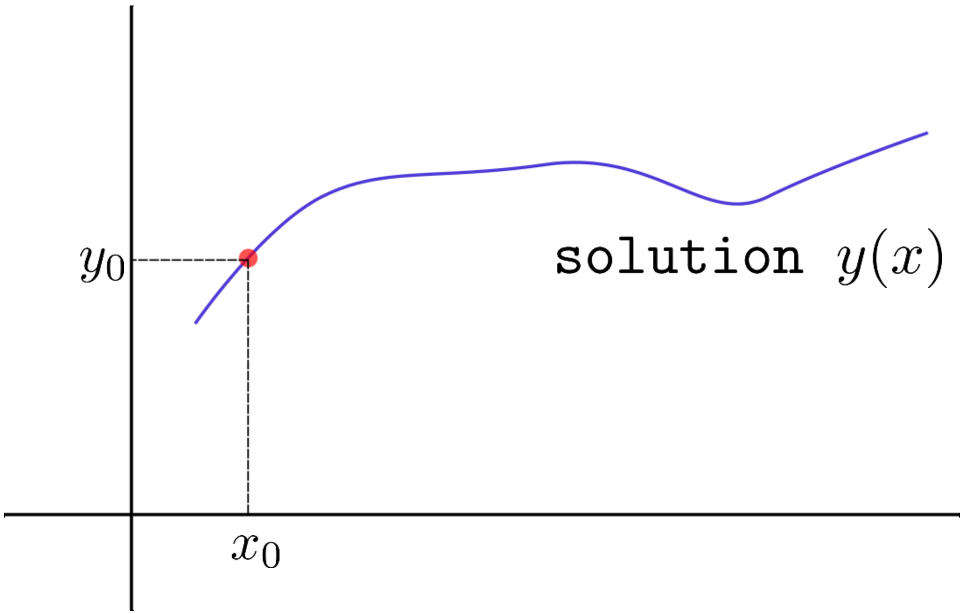
In practice, however, you will deal most of the time with DEs which can not be solved analytical. The only way to solve these DEs is using the computer (numerically).

We start with the most simple method, the so-called Euler method. All better methods are improvements of this method.

Euler method

We are going to solve the differential equation $\dfrac{dy}{dx} = f(x, y)$

Suppose we know a point through which the solution goes (in practice this is mostly the initial condition).

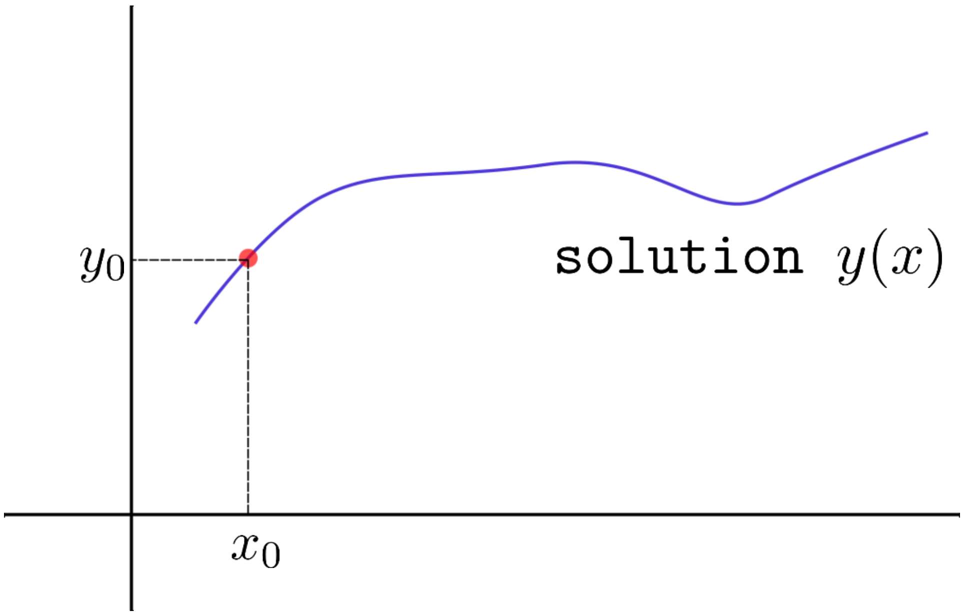


Euler method

We are going to solve the differential equation $\dfrac{dy}{dx} = f(x, y)$

Suppose we know a point through which the solution goes (in practice this is mostly the initial condition).

We also know the slope in this point; it is $f(x_0, y_0)$



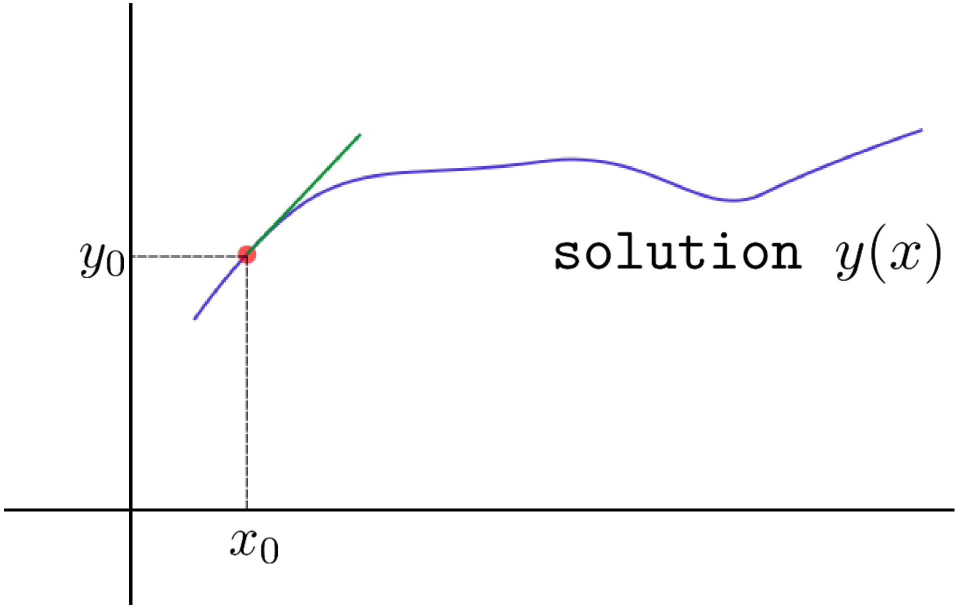
Euler method

We are going to solve the differential equation $\frac{dy}{dx} = f(x, y)$

Suppose we know a point through which the solution goes (in practice this is mostly the initial condition).

We also know the slope in this point, it is $f(x_0, y_0)$

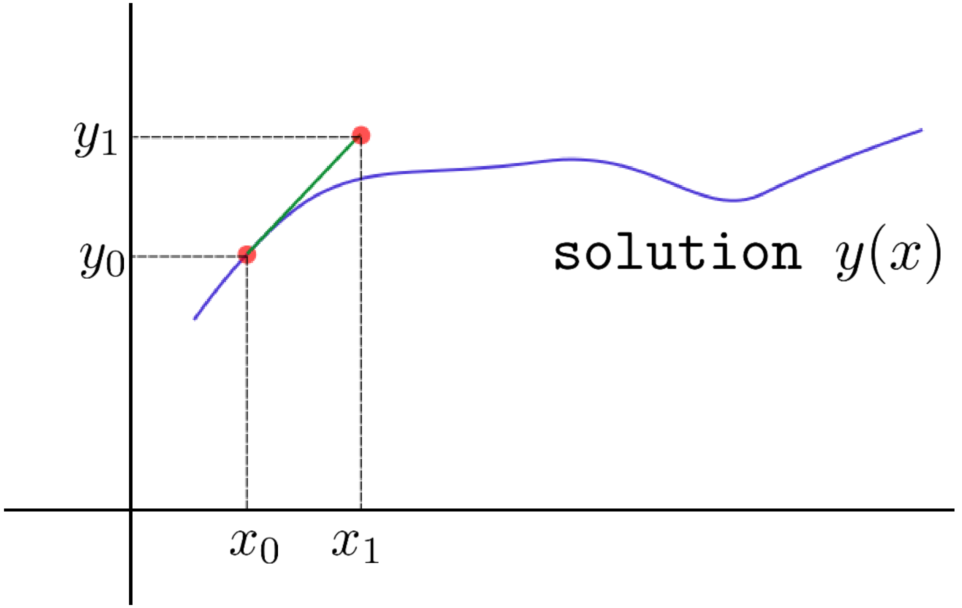
In the vicinity of this point, the tangent lies close to the real solution.



Euler method

We are going to solve the differential equation $\dfrac{dy}{dx} = f(x,y)$

Suppose we know a point through which the solution goes (in practice this is mostly the initial condition).



We also know the slope in this point, it is $f(x_0, y_0)$

In the vicinity of this point, the tangent lies close to the real solution.

A little bit further away the solution can be approximated by $y_1 = y_0 + f(x_0, y_0)\Delta x$

Euler method

To calculate the next point, we will act in the same way.

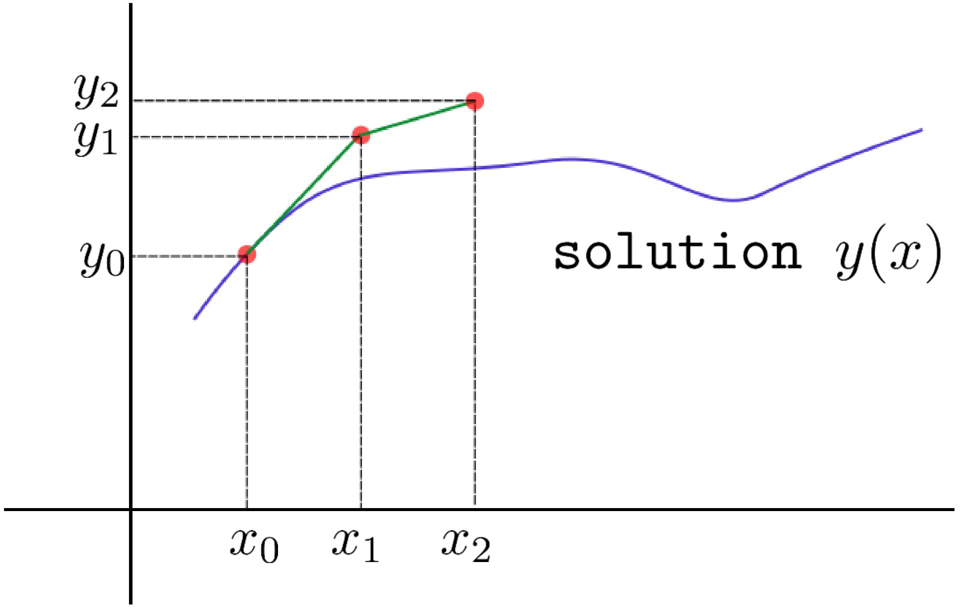
The slope in the point (x_1, y_1) is $f(x_1, y_1)$

Hence, the next approximation is:

$$y_2 = y_1 + f(x_1, y_1)\Delta x$$

Each following point is calculated in the same way. This method can be programmed easily (for loop):

$$y_{n+1} = y_n + f(x_n, y_n)\Delta x$$



Note that we actually use the definition of the derivation, but without using the limit.

$$f(x,y) = \lim_{\Delta x \to 0} \frac{y(x+\Delta x) - y(x)}{\Delta x} = \frac{dy}{dx}$$

Kom verder

SAXION

Example: $\dfrac{dy}{dx} = -y \quad y(0) = 5$

$$y_{n+1} = y_n + f(x_n, y_n)\Delta x$$

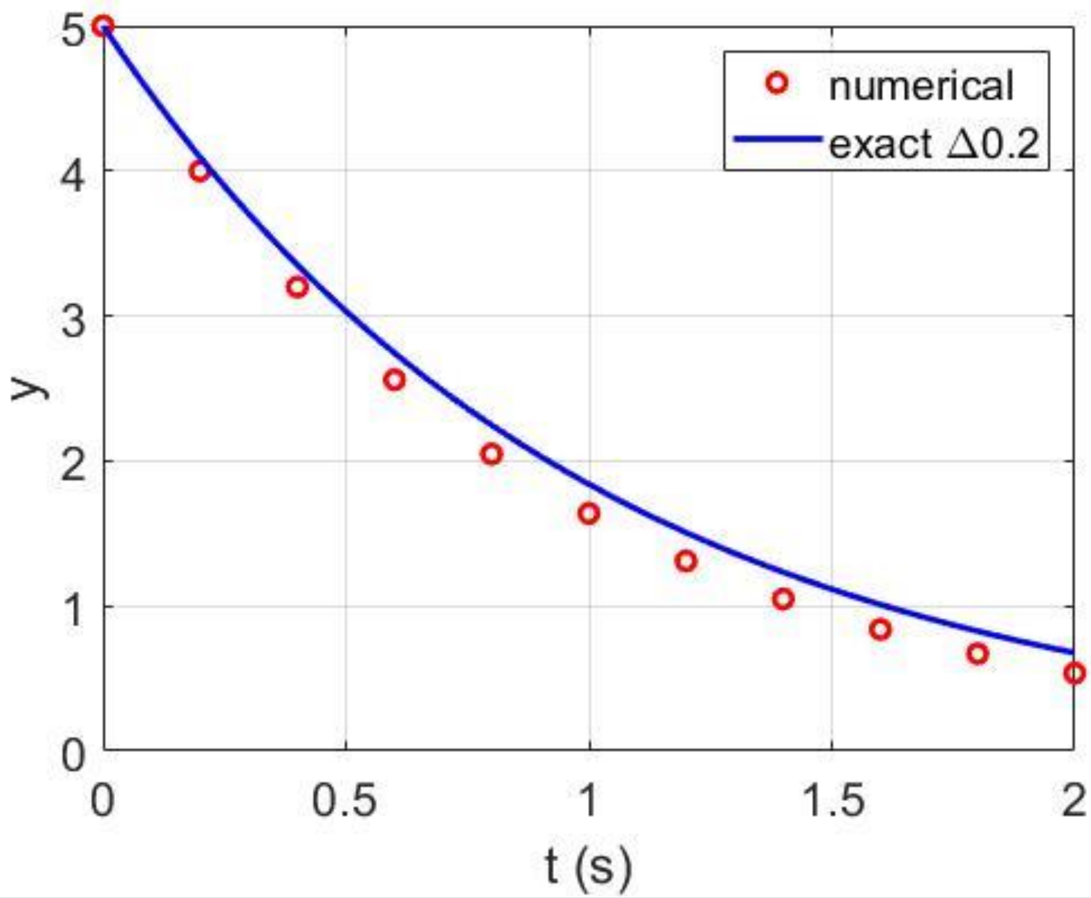
The first few steps

Step 'k'	x(k)	y(k)	f(x(k),y(k))	Δx= 0.2	f(x(k),y(k)) Δx
	0	5	-5	0.2	-1
1	0.2	5 + (-1) = 4	-4	0.2	-0.8
2	0.4	4 + (-0.8) = 3.2	-3.2	0.2	-0.64
3	0.6	3.2 + (-0.64) = 2.56			

Example: $\frac{dy}{dx} = -y \quad y(0) = 5$

Analytical solution $y(x) = 5e^{-x}$

Analytical Solution	Euler
5.000	5.000
4.094	4.000
3.352	3.200
2.744	2.560
2.247	2.048
1.839	1.638
1.506	1.311
1.233	1.049
1.009	0.839
0.826	0.671
0.677	0.537

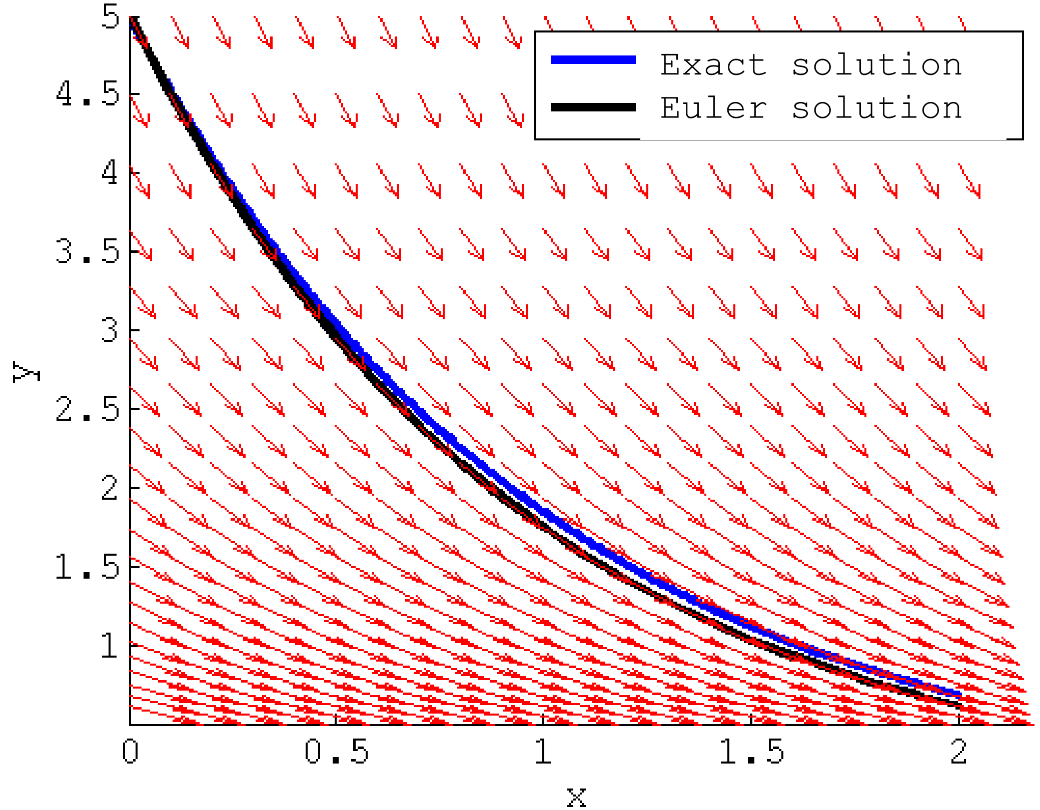


Numerical methods always give an approximation of the exact solution

Kom verder

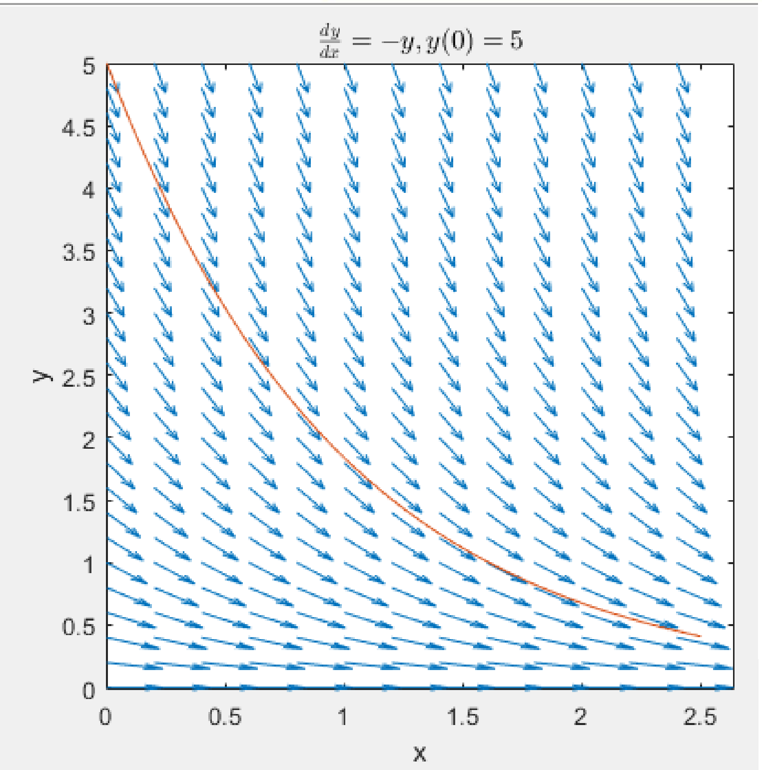
SAXION

Example: $\frac{dy}{dx} = -y \quad y(0) = 5$



The Euler solution follows a directional field. Hence, it is always a good idea to plot the directional field together with the numerical solution, as a qualitative check.

Intermezzo: direction fields (extra)




```
% dy/dx = -y, y(0)=5
% S slope, L length
[x,y]=meshgrid(0:.2:2.5,-0:.2:5);
S= -y;
L=sqrt(1+S.^2);
quiver(x,y,1./L,S./L);
axis tight square
hold on
f=@(u,v) -v; % use u & v i.s.o. x & y.
% ode45 is an advanced numerical method
[u,v]=ode45(f,[0,2.5], 5); % range of u is [0,2.5], u(0)= 5
plot(u,v)
xlabel ('x')
ylabel ('y')
title('$\frac{dy}{dx} = -y, y(0)=5$','Interpreter','latex')
```

The built-in function `ode45` is an advanced numerical method based on a explicit Runge-Kutta (4,5) formula. In real (matlab) live, you will use it i.s.o an Euler or Heun method. For educational purposes, however we first will look at the Euler and Heun method.

Kom verder

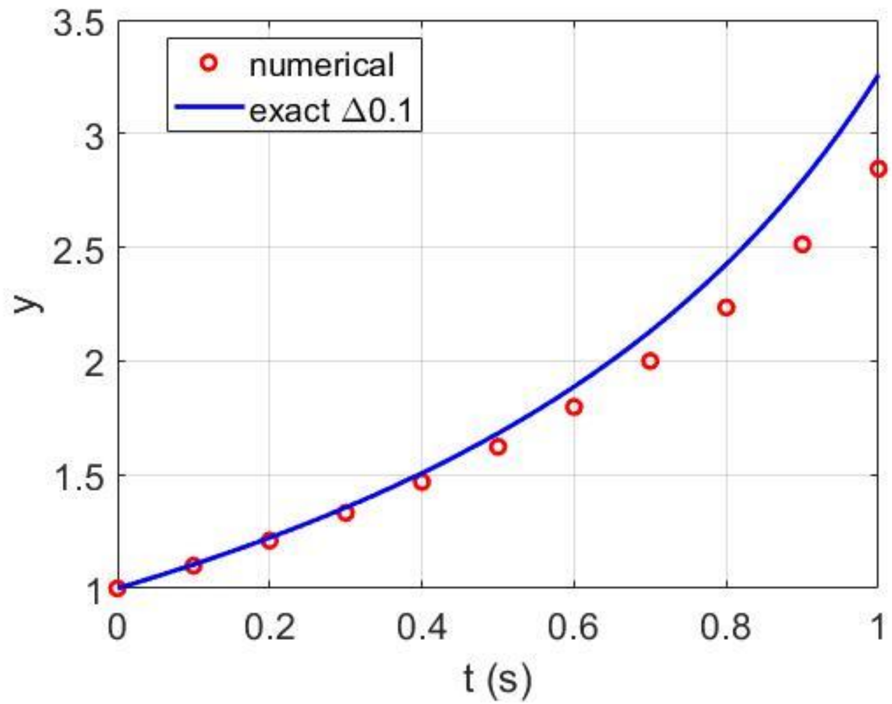
SAXION

Example: $\dfrac{dy}{dx} = \dfrac{y^2}{x+1}, \quad y(0) = 1$

Analytical solution

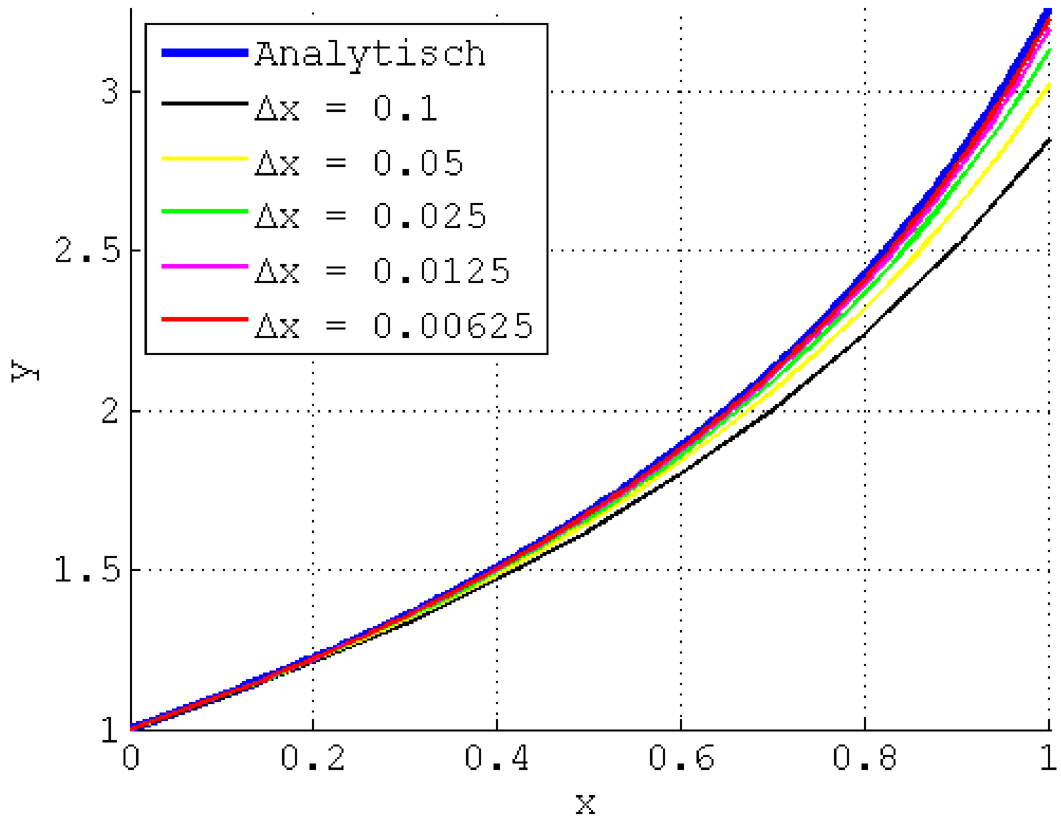
$$y(x) = \frac{1}{1 - \ln(x+1)}$$

x	Euler: Y	Analytisch: y(x)	fout |Y - y(x)|
0.0000	1.0000	1.0000	0.0000
0.1000	1.1000	1.1054	0.0054
0.2000	1.2100	1.2230	0.0130
0.3000	1.3320	1.3557	0.0237
0.4000	1.4685	1.5071	0.0386
0.5000	1.6225	1.6820	0.0595
0.6000	1.7980	1.8868	0.0888
0.7000	2.0001	2.1305	0.1304
0.8000	2.2354	2.4259	0.1905
0.9000	2.5130	2.7922	0.2791
1.0000	2.8454	3.2589	0.4135



The error typically increases after each following step.

How can we get a better numerical approximation?
Make the time step smaller.



The numerical solution comes closer and closer to the analytical solution.
But how decreases the error as function of the step size?

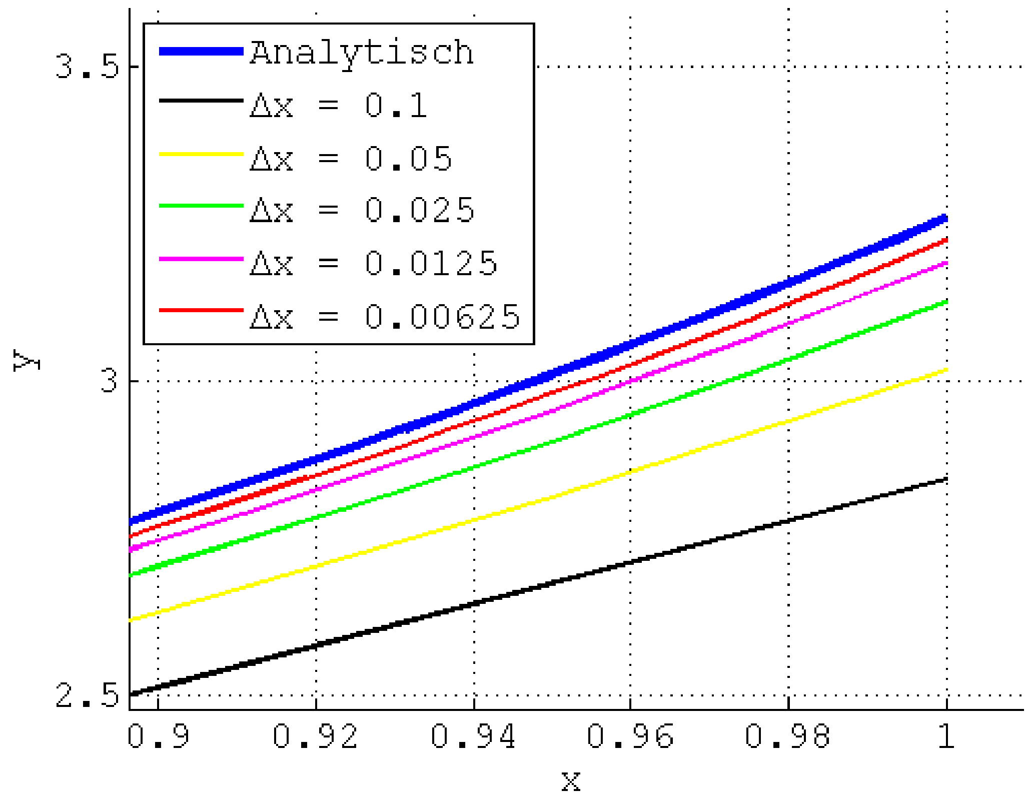
How does the error scale?

The error seems roughly to **halve** when the time step is halved.

We calculate the error at $x = 1$ by: $E = |y_{\text{an}}(1) - y_{\Delta x}(1)|$

For different step sizes we get the following values:

Step size	Error
0.1000	0.4135
0.0500	0.2408
0.0250	0.1316
0.0125	0.0690
0.0063	0.0354



How does the error scale?

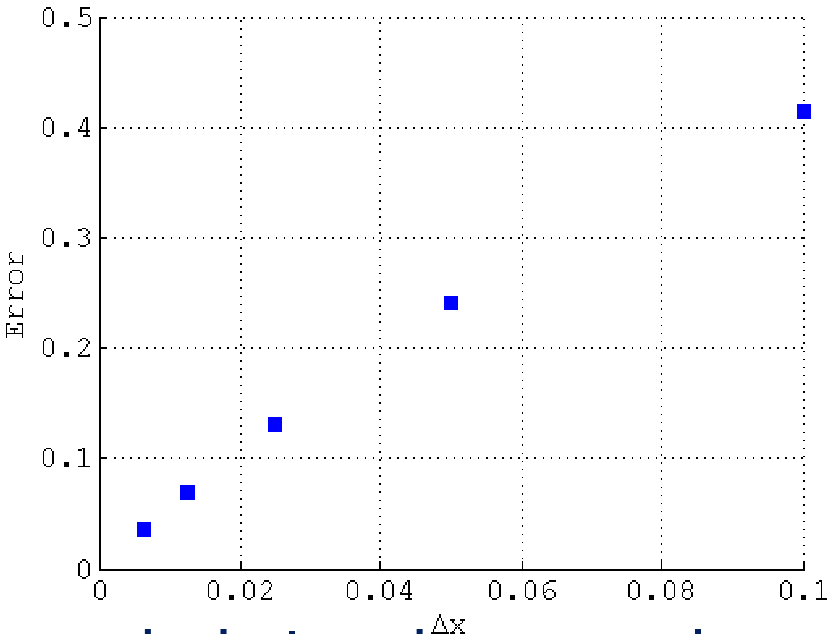
If we divide the errors, we see that these roughly decrease with a factor 2.

$$\frac{E_{\Delta x=0.05}}{E_{\Delta x=0.1}} \rightarrow 0.5824$$

$$\frac{E_{\Delta x=0.025}}{E_{\Delta x=0.05}} \rightarrow 0.5462$$

$$\frac{E_{\Delta x=0.0125}}{E_{\Delta x=0.025}} \rightarrow 0.5247$$

$$\frac{E_{\Delta x=0.00625}}{E_{\Delta x=0.0125}} \rightarrow 0.5128$$



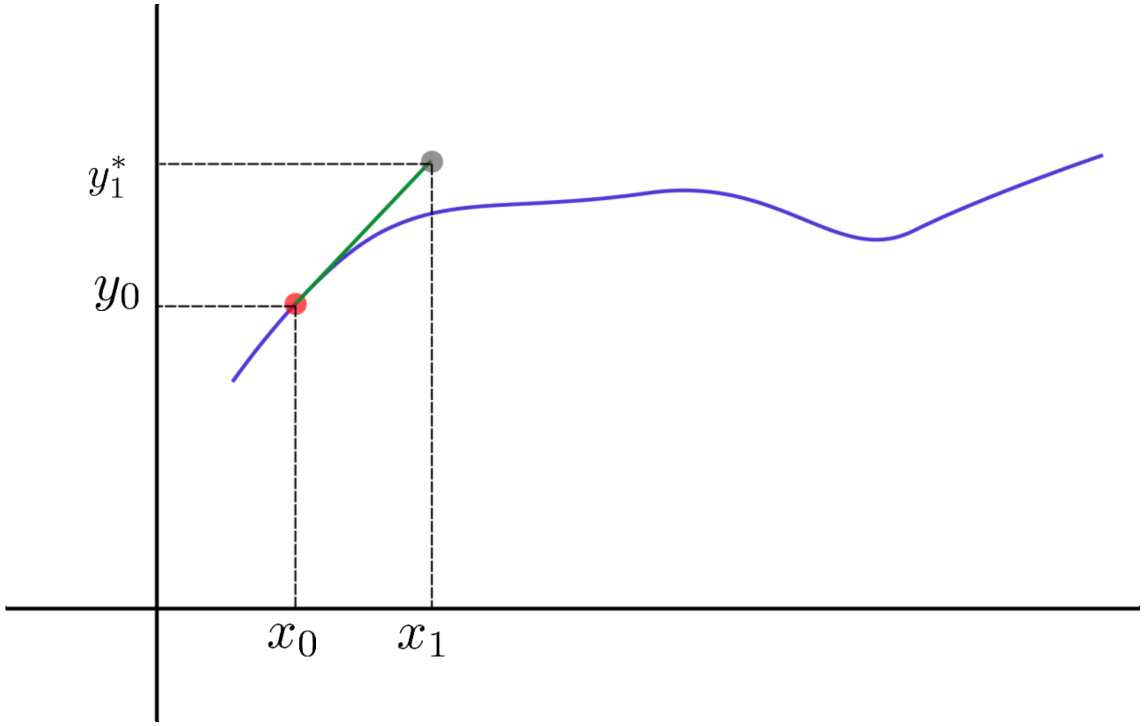
The Euler method is called a 1st order method, since the error scales according to: $E \propto \Delta x$
Note: this name has NOTHING to do with the fact that we are solving 1st order DEs.

Advantages and disadvantages of the Euler method

- An advantage of a 1^{st} order method is that it is very fast (to calculate). For the next value only 1 calculation is needed at the right side of the DE.
- A disadvantage is that a 1^{st} order method is not very precise.
- In general you need a small step size to increase the precision. That means that a lot of steps have to be taken, which will decrease the speed.

Improved Euler method (Heun method)

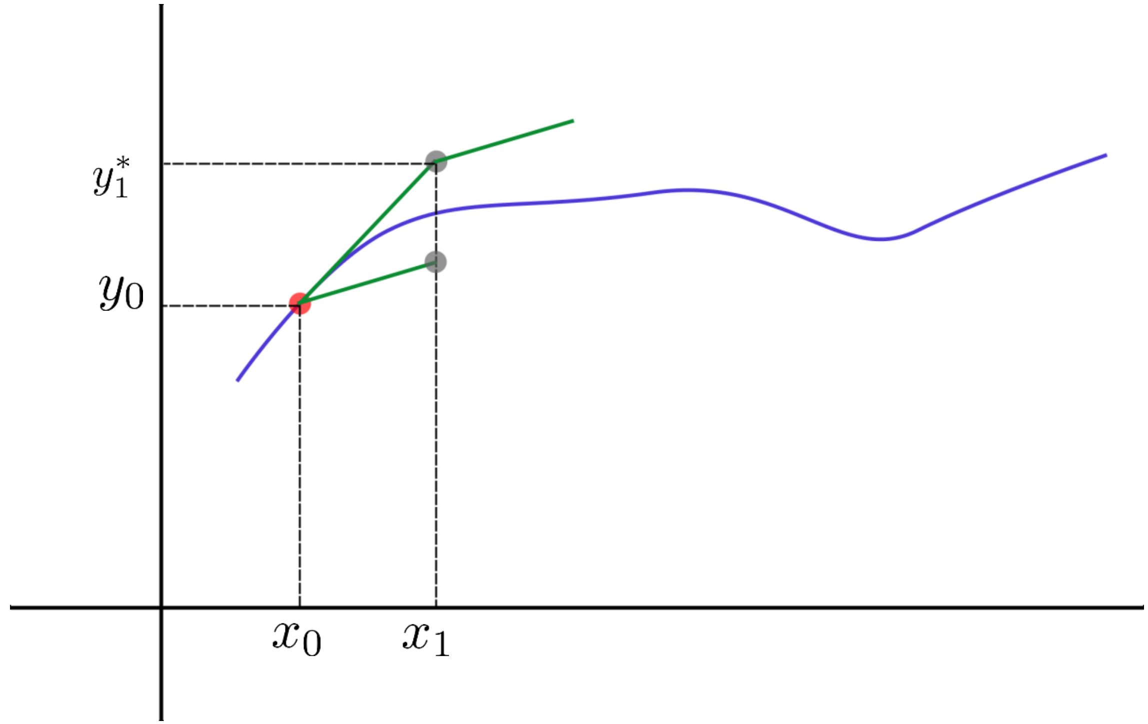
We will get a better method, if we will make a better choice for the slope. Using the Euler method the slope was taken in the point (x_0, y_0). So it was assumed that this slope would not change in the interval $[x_0, x_1]$. In the figure below, you see that we will get a too high value if we use the Euler method.



Improved Euler method (Heun method)

In the point (x_1, y_1^*) the slope will be already somewhat lower. If we would use that slope, our estimation would be too low.

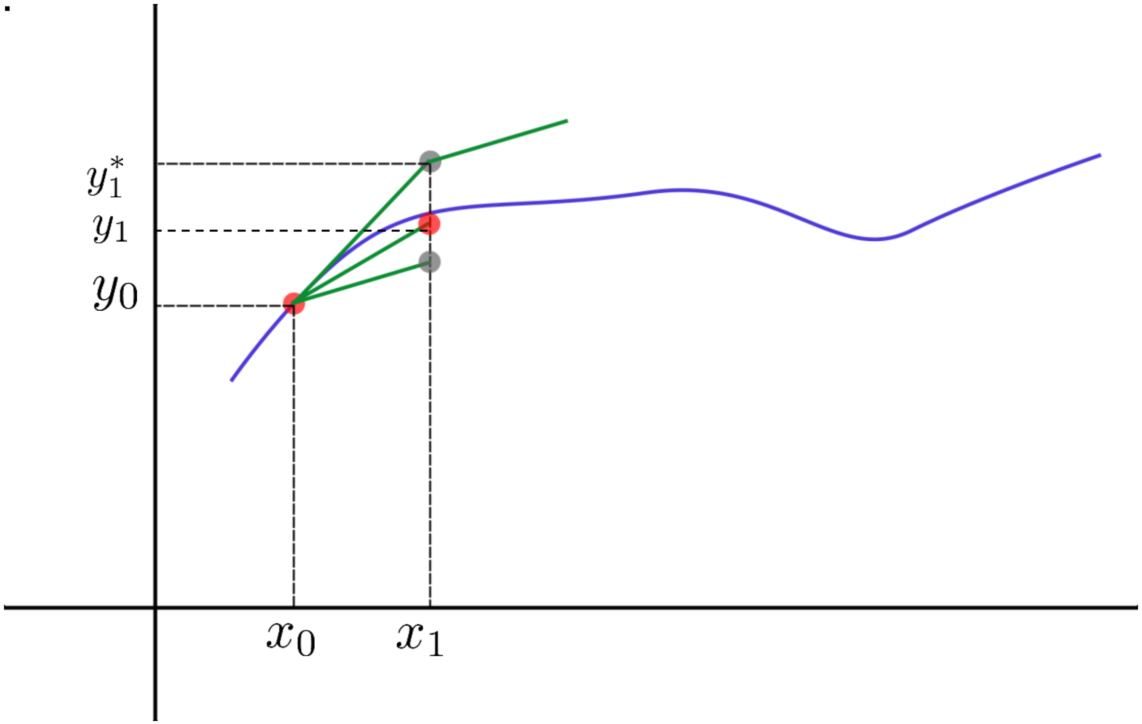
The point (x_1, y_1^*) does not lay on the function, however, we can use the equation $y\prime = f(x, y)$ to calculate its slope.



Improved Euler method (Heun method)

In the point (x_1, y_1) the slope will be already somewhat lower. If we would use this slope, your approximation is a little bit to low. However, the average of the two grey point is close to the real solution. This is the essence of the improved Euler method. So we take the average of the slopes in the points (x_0, y_0) and (x_1, y_1^*) :

$$y_1 = y_0 + \frac{f(x_0, y_0) + f(x_1, y_1^*)}{2}\Delta x$$



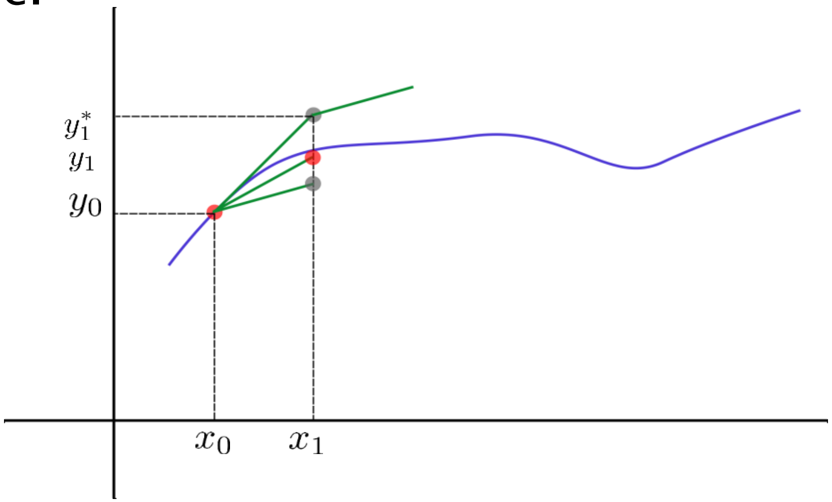
Improved Euler method (Heun method)

The Heun method needs the slope of the unknown point (x_1, y_1^*)

You can obtain an approximation for this slope by executing an Euler step.

So the algorithm is as follows

1. Execute a standard Euler step. This will give you an approximation for y_1^*. This is a temporary value, which we will use to obtain an improved approximation of the slope.
2. Calculate the average of the slopes in the points (x_0, y_0) and (x_1, y_1^*) and use this average to calculate the final value for y_1



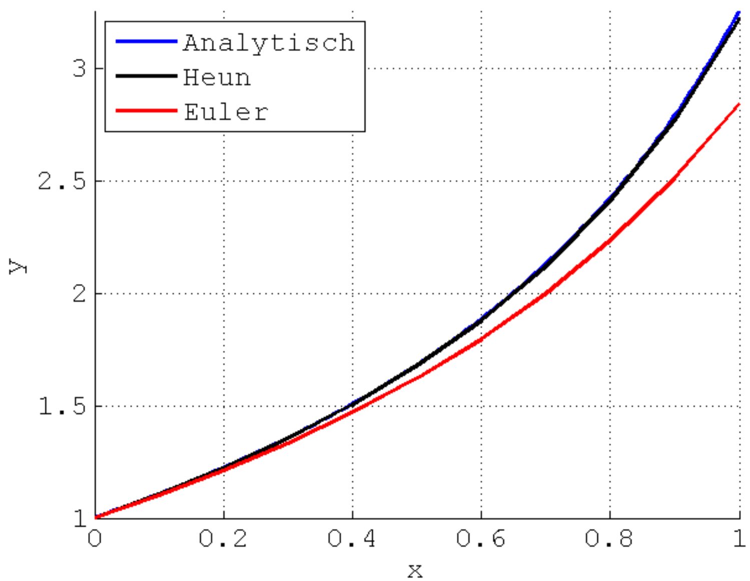
Comparison between the Euler and the Heun methods

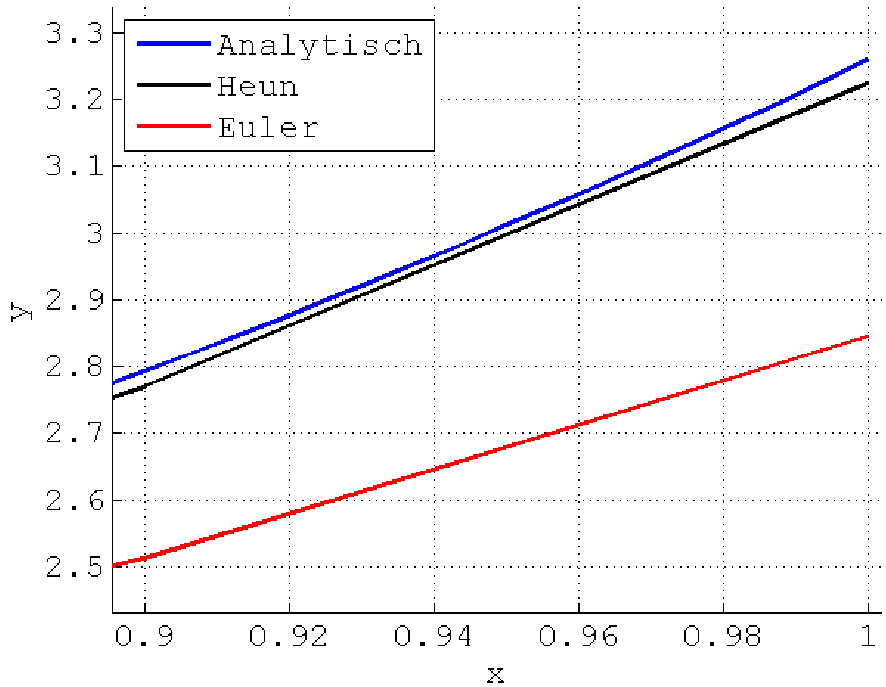
We will take again the DE:

$$\frac{dy}{dx} = \frac{y^2}{x+1}, \quad y(0) = 1$$

with analytical solution

$$y(x) = \frac{1}{1 - \ln(x+1)}$$





Step size used at both methods: $\Delta x = 0.1$

Notice the great improvement

Comparison between the Euler and the Heun methods

x	Analytisch	Euler	Heun	Error Euler	Error Heun
0.0000	1.0000	1.0000	1.0000	0.0000	0.0000
0.1000	1.1054	1.1000	1.1050	0.0054	0.0004
0.2000	1.2230	1.2100	1.2221	0.0130	0.0009
0.3000	1.3557	1.3320	1.3541	0.0237	0.0016
0.4000	1.5071	1.4685	1.5044	0.0386	0.0027
0.5000	1.6820	1.6225	1.6778	0.0595	0.0042
0.6000	1.8868	1.7980	1.8804	0.0888	0.0064
0.7000	2.1305	2.0001	2.1208	0.1304	0.0097
0.8000	2.4259	2.2354	2.4111	0.1905	0.0148
0.9000	2.7922	2.5130	2.7693	0.2791	0.0229
1.0000	3.2589	2.8454	3.2228	0.4135	0.0361

Using the same step size the Heun Error is much smaller then the Euler Error. However, the error typically grows at each next step.

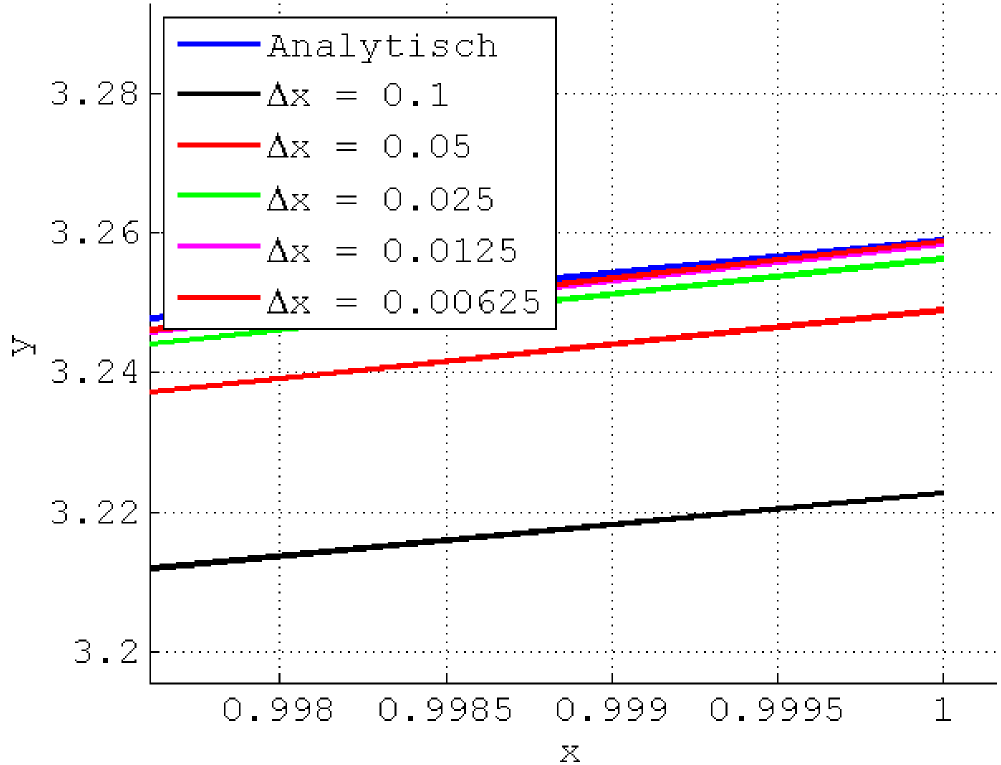
How does the error scale at the Heun method?

The error seems roughly to become **4x smaller** when the step size is halved.

We calculate the error at $x = 1$ bv: $E = |y_{\mathrm{an}}(1) - y_{\Delta x}(1)|$

For different step sizes we get the following values:

Step size	Error
0.1000	0.0361
0.0500	0.0099
0.0250	0.0026
0.0125	0.0007
0.0063	0.0002



Convergence of the Heun method

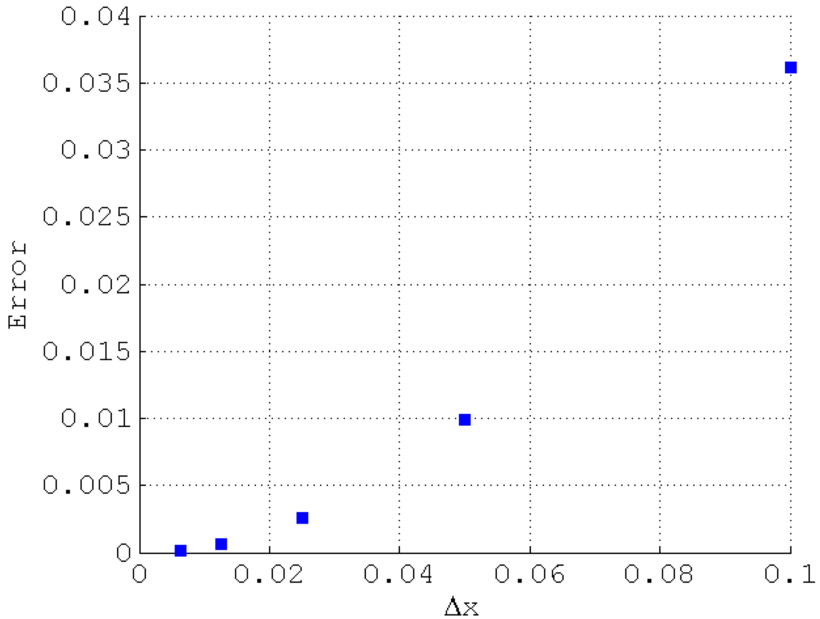
Just like the Euler method, we can investigate how the error scales as function of the step size. We again will place the ratios of the errors under each other.

$$\frac{E_{\Delta x=0.05}}{E_{\Delta x=0.1}} \rightarrow 0.2745$$

$$\frac{E_{\Delta x=0.025}}{E_{\Delta x=0.05}} \rightarrow 0.2610$$

$$\frac{E_{\Delta x=0.0125}}{E_{\Delta x=0.025}} \rightarrow 0.2552$$

$$\frac{E_{\Delta x=0.00625}}{E_{\Delta x=0.0125}} \rightarrow 0.2525$$



The Heun method is called a 2nd order method, since the error scales according to: $E \propto (\Delta x)^2$

Note: this name has NOTHING to do with the order of the DEs we are solving.

Kom verder

Advantages and disadvantages of the Heun method

- An advantage is that a 2nd order method is very precise. In general you can take larger steps for the same precision. That means that much less steps are needed compared to a first order method.

- A disadvantage of the Heun method is that for each update 2 calculations of the right side of the DE are needed. Often the evaluation of this function will actually cost most of the computing time. For more complicated functions the advantage of a larger step size could be neutralised.

Kom verder

SAXION

Even better methods: Runge-Kutta methods

By making even better approximations for the slope, we can obtain even more precise numerical results. The standard method, which is used often in practice is the RK4 method (this is an abbreviation for Runge-Kutta 4th order). The error in this order scales with the 4th power of the step size. This means that when halving the step size, the error will become a factor 16 smaller. A disadvantage is that for each update with the RK4 method, the slope has to be evaluated 4 times. In Matlab an advanced RK4 method is available.

Questions