Math 2650

A. J. Meir


```
> restart:with(DEtools):with(plots):
```

Euler's method

We now look at a very simple numerical method for approximating solutions of differential equations. It is essentially based on the direction field, and an approximation of the derivative as a difference quotient.
Consider the equation

$$\frac{dx}{dt} = x^2 - t^2 \qquad x(0) = \frac{1}{2}$$

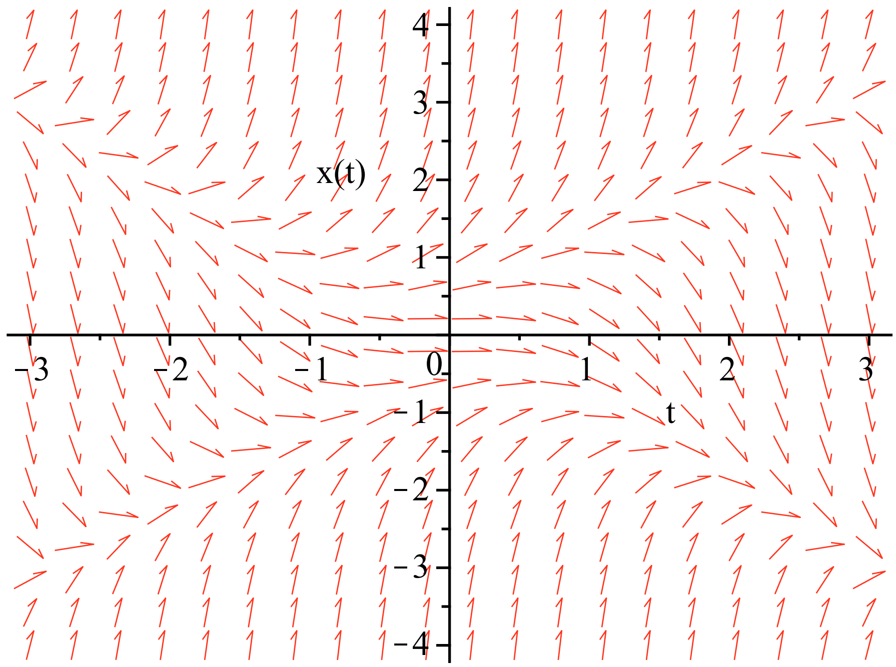
Lets try and solve the equation

```
> dsolve({diff(x(t),t)=x(t)^2-t^2,x(0)=1/2},x(t));
```

$$x(t) = \begin{cases} -\frac{t\left(\frac{1}{4}\,\frac{\pi\left(2\,\Gamma\left(\frac{3}{4}\right)^2\sqrt{2}+\pi\right)\mathrm{BesselI}\left(-\frac{3}{4},\frac{1}{2}\,t^2\right)}{\Gamma\left(\frac{3}{4}\right)^2} - \mathrm{BesselK}\left(\frac{3}{4},\frac{1}{2}\,t^2\right)\right)}{\frac{1}{4}\,\frac{\pi\left(2\,\Gamma\left(\frac{3}{4}\right)^2\sqrt{2}+\pi\right)\mathrm{BesselI}\left(\frac{1}{4},\frac{1}{2}\,t^2\right)}{\Gamma\left(\frac{3}{4}\right)^2} + \mathrm{BesselK}\left(\frac{1}{4},\frac{1}{2}\,t^2\right)} & t < 0 \\ \frac{1}{2} & t = 0 \\ -\frac{t\left(-\frac{1}{4}\,\frac{\pi\left(-2\,\Gamma\left(\frac{3}{4}\right)^2\sqrt{2}+\pi\right)\mathrm{BesselI}\left(-\frac{3}{4},\frac{1}{2}\,t^2\right)}{\Gamma\left(\frac{3}{4}\right)^2} - \mathrm{BesselK}\left(\frac{3}{4},\frac{1}{2}\,t^2\right)\right)}{-\frac{1}{4}\,\frac{\pi\left(-2\,\Gamma\left(\frac{3}{4}\right)^2\sqrt{2}+\pi\right)\mathrm{BesselI}\left(\frac{1}{4},\frac{1}{2}\,t^2\right)}{\Gamma\left(\frac{3}{4}\right)^2} + \mathrm{BesselK}\left(\frac{1}{4},\frac{1}{2}\,t^2\right)} & 0 < t \end{cases}$$

This is a mess so lets look at the direction field to get some insight into the problem

```
> DF:=DEplot(diff(x(t),t)=x(t)^2-t^2,x(t),
  t=-3..3,x=-4..4):
> display(DF);
```



Lets try and use Euler's method to construct an approximate solution

Below we write a little Maple procedure for the forward Euler's method.

Euler's Method

```
> FEuler:=proc(f,t0,x0,h,N)
  local t,x,n,sol:
  t:=evalf(t0):
  x:=evalf(x0):
  sol:=[[t,x]]:
  for n from 1 to N do
  x:=x+h*f(t,x):
  t:=t+h:
  sol:=[op(sol),[t,x]]:
  od;
  end;
```

FEuler := **proc**(f, t0, x0, h, N)

local t, x, n, sol;

t := evalf(t0);

x := evalf(x0);

sol := [[t, x]];

for n **to** N **do**

$x := x + h*f(t, x);\ t := t + h;\ sol := [op(sol), [t, x]]$

end do

end proc

```
> t0:=0;tf:=3;x0:=1/2;
```

$$t0 := 0$$

$$tf := 3$$

$$x0 := \frac{1}{2}$$

```
> N:=3;
```

$$N := 3$$

```
> h:=evalf((tf-t0)/N);
```

$$h := 1.$$

```
> f:=(t,x)->x^2-t^2;
```

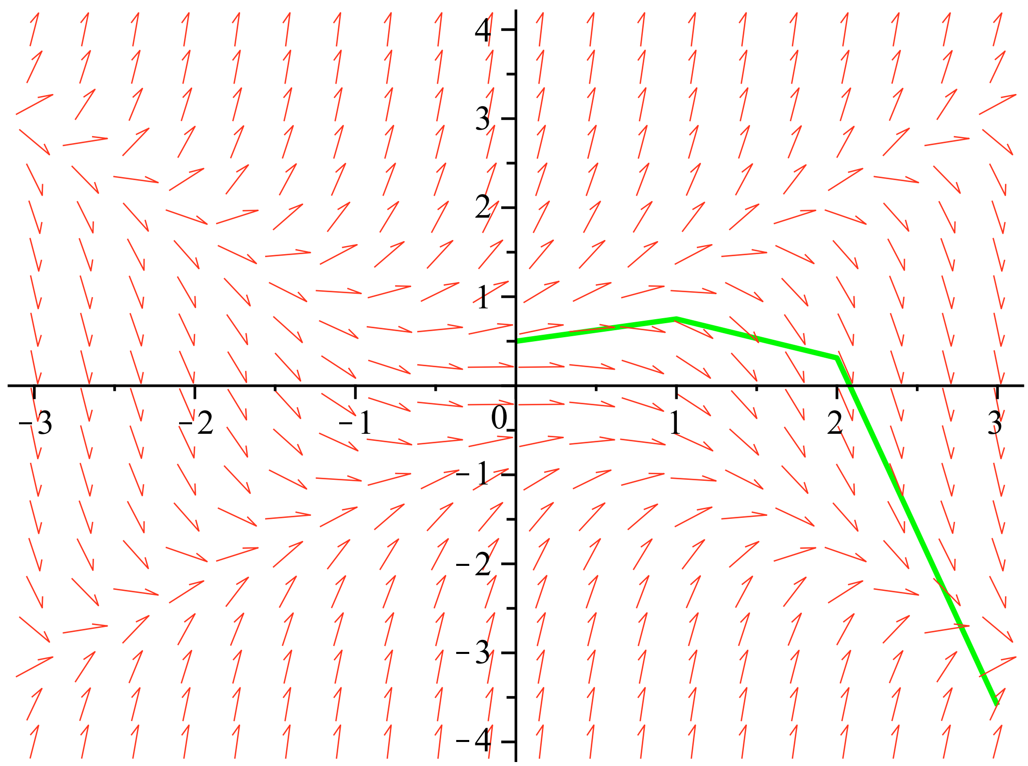
$$f := (t, x) \to x^2 - t^2$$

```
> out:=FEuler(f,t0,x0,h,N);
```

$out := [[0., 0.5000000000], [1., 0.7500000000], [2., 0.3125000000], [3., -3.589843750]]$

```
> EulerApprox:=plot(out,color=green,thickness=2):
```

```
> display(EulerApprox,DF);
```

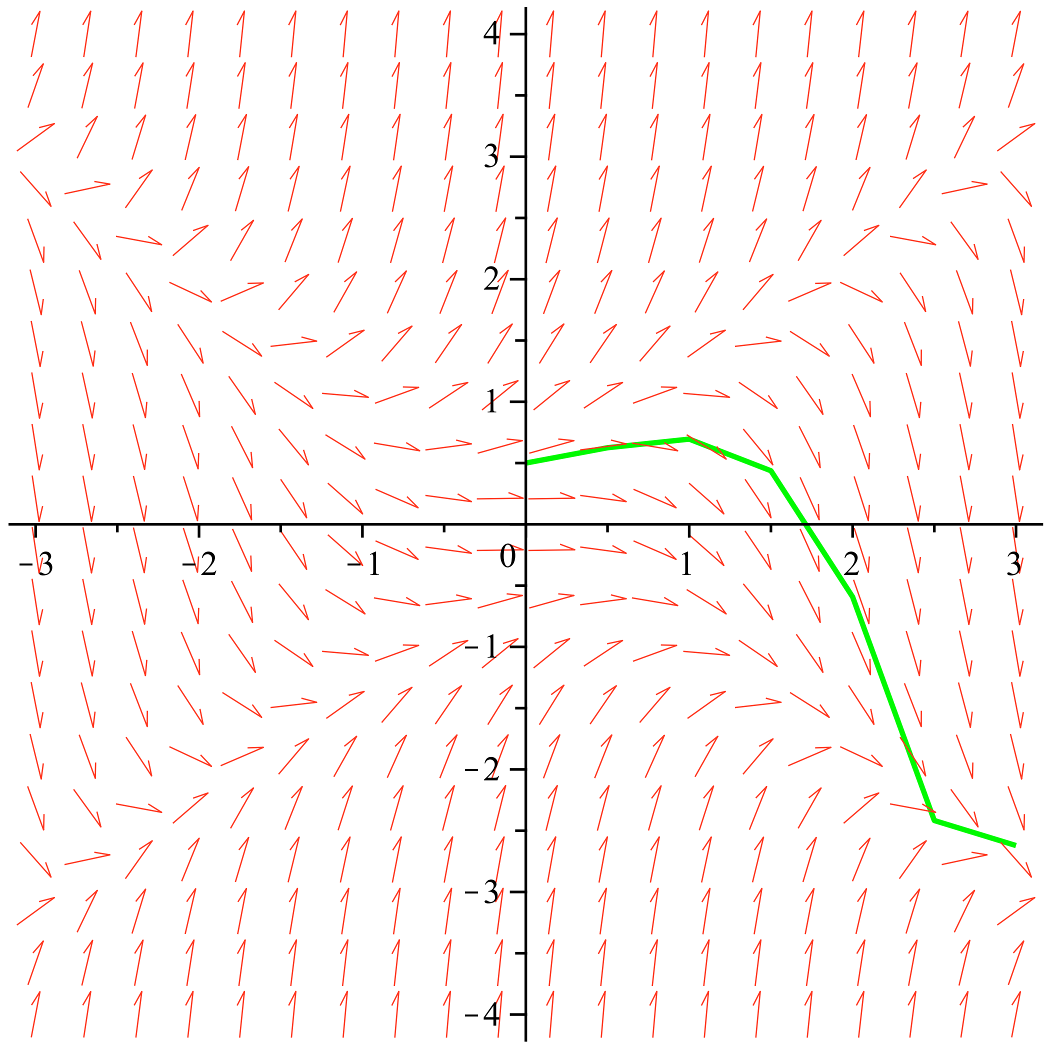



```
> N:=6;
> h:=evalf((tf-t0)/N);
> out:=FEuler(f,t0,x0,h,N);
> EulerApprox:=plot(out,color=green,thickness=2):
> display(EulerApprox,DF);
```

$$N := 6$$

$$h := 0.5000000000$$

$out := [[0., 0.5000000000], [0.5000000000, 0.6250000000], [1.000000000, 0.6953125000], [1.500000000, 0.4370422364], [2.000000000, -0.5924548056], [2.500000000, -2.416953458], [3.000000000, -2.621121449]]$

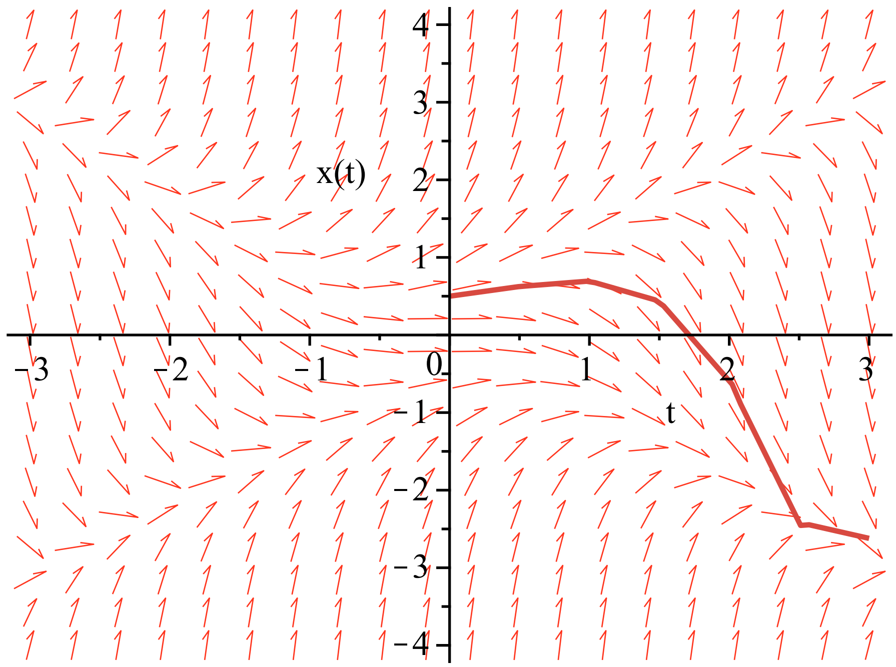


We can also use Maple's built in numerical o.d.e. "solvers", but then we may not have precise control over the step size.

```
> ans:=dsolve({diff(x(t),t)=x(t)^2-t^2,x(0)=1/2},x(t),numeric,
  method=classical[foreuler],stepsize=h);
```

$$ans := \mathbf{proc}(x\_classical) \ \ldots \ \mathbf{end\ proc}$$

```
> approx:=odeplot(ans,[t,x(t)],0..3,color=orange,
  thickness=2):
> display(DF,approx);
```



We now look at a very simple stiff o.d.e. Consider

$$\frac{dx}{dt} = -10\,x \qquad x(0) = 10$$

```
> t0:=0;tf:=3;x0:=10;
> N:=10;
> h:=evalf((tf-t0)/N);
> f:=(t,x)->-10*x;
> out:=FEuler(f,t0,x0,h,N);
> EulerApprox:=plot(out,color=green,thickness=2):
> DF:=DEplot(diff(x(t),t)=-10*x(t),x(t),
  t=-1..3,x=-10..10):
> display(EulerApprox,DF);
```

$$t0 := 0$$

$$tf := 3$$

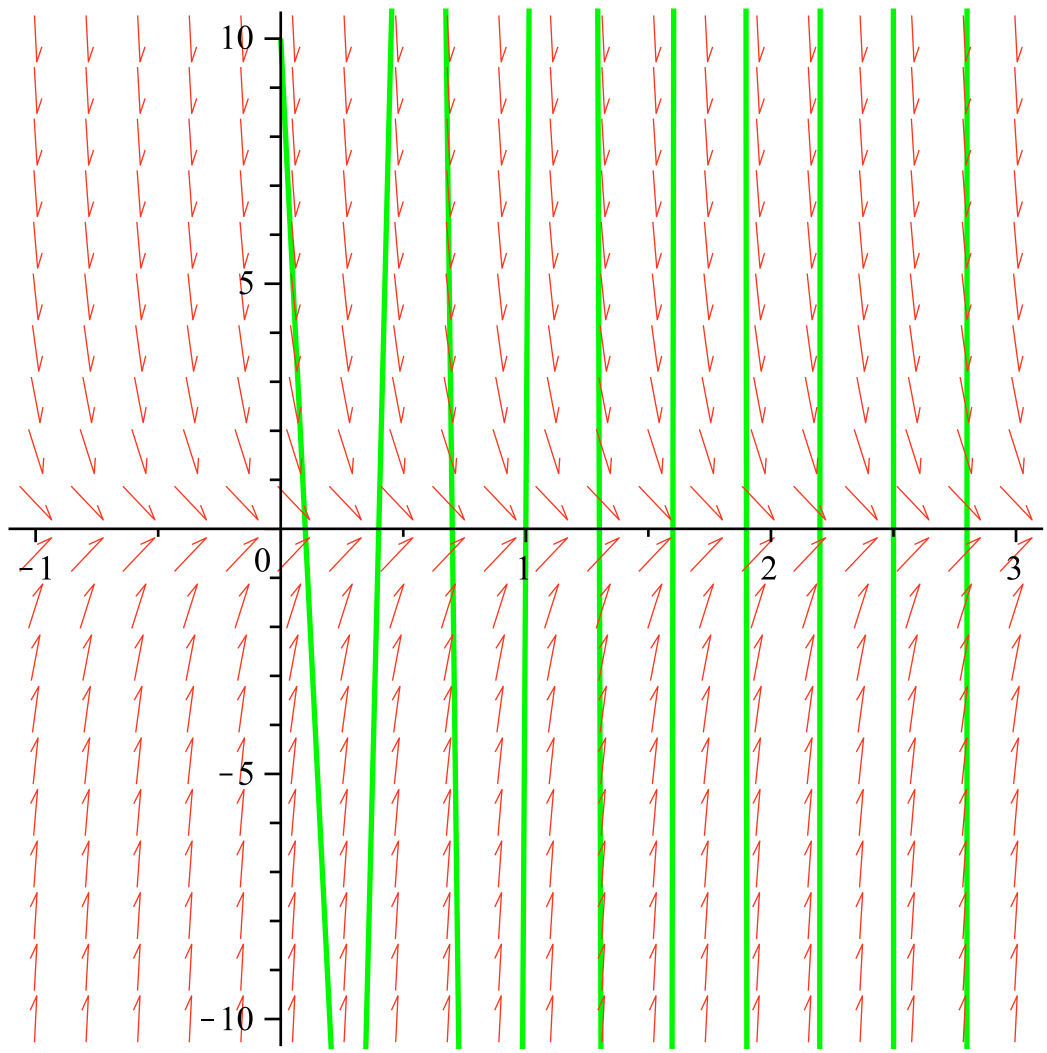
$$x0 := 10$$

$$N := 10$$

$$h := 0.3000000000$$

$$f := (t, x) \rightarrow -10\,x$$

out := [[0., 10.], [0.3000000000, -20.00000000], [0.6000000000, 40.00000000], [0.9000000000, -80.00000000], [1.200000000, 160.0000000], [1.500000000, -320.0000000], [1.800000000, 640.0000000], [2.100000000, -1280.000000], [2.400000000, 2560.000000], [2.700000000, -5120.000000], [3.000000000, 10240.00000]]

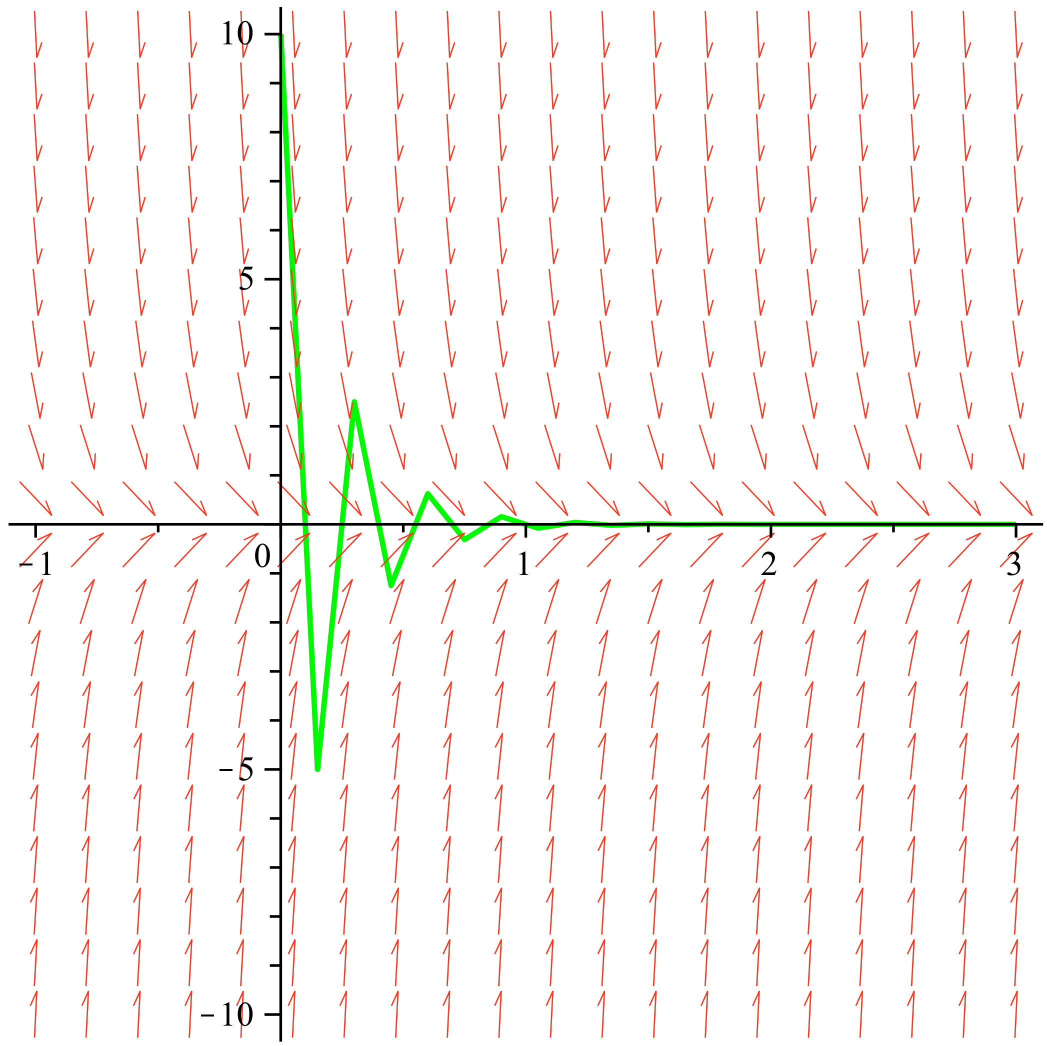



```
> N:=20;
> h:=evalf((tf-t0)/N);
> out:=FEuler(f,t0,x0,h,N):
> EulerApprox:=plot(out,color=green,thickness=2):

> DF:=DEplot(diff(x(t),t)=-10*x(t),x(t),
  t=-1..3,x=-10..10):
> display(EulerApprox,DF);
```

$$N := 20$$

$$h := 0.1500000000$$

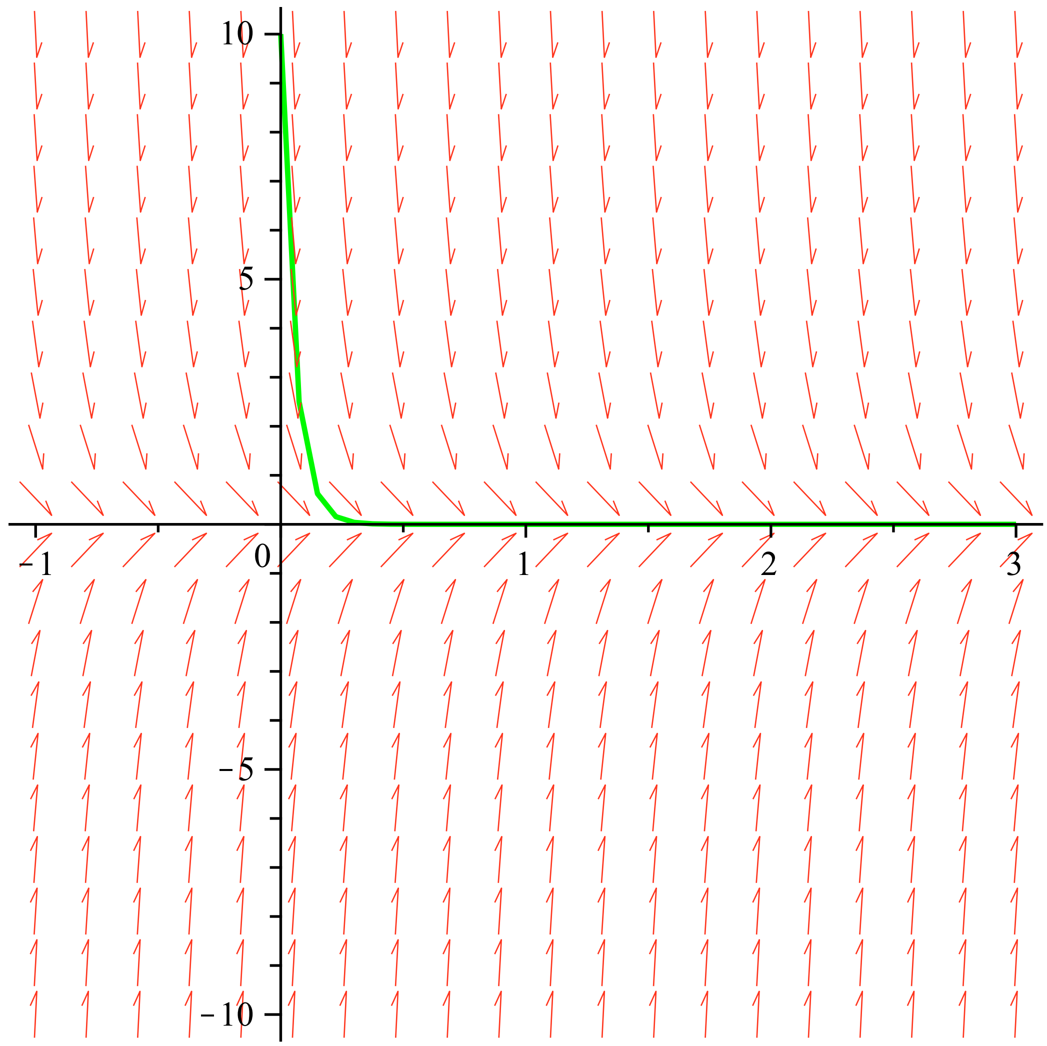



```
> N:=40;
> h:=evalf((tf-t0)/N);
> out:=FEuler(f,t0,x0,h,N):
> EulerApprox:=plot(out,color=green,thickness=2):

> DF:=DEplot(diff(x(t),t)=-10*x(t),x(t),
  t=-1..3,x=-10..10):
> display(EulerApprox,DF);
```

$$N := 40$$

$$h := 0.07500000000$$



10
5
0
-5
-10
-1
1
2
3