

Numerische Analysis $x + \cos(x)$

Prof. Dr. Dörte Haftendorn: Mathematik mit MuPAD 4, Update Mai 07
existiert auch als MuPAD 3-Version Aug. 05

www.uni-lueneburg.de/mathe-lehramt

<http://haftendorn.uni-lueneburg.de>

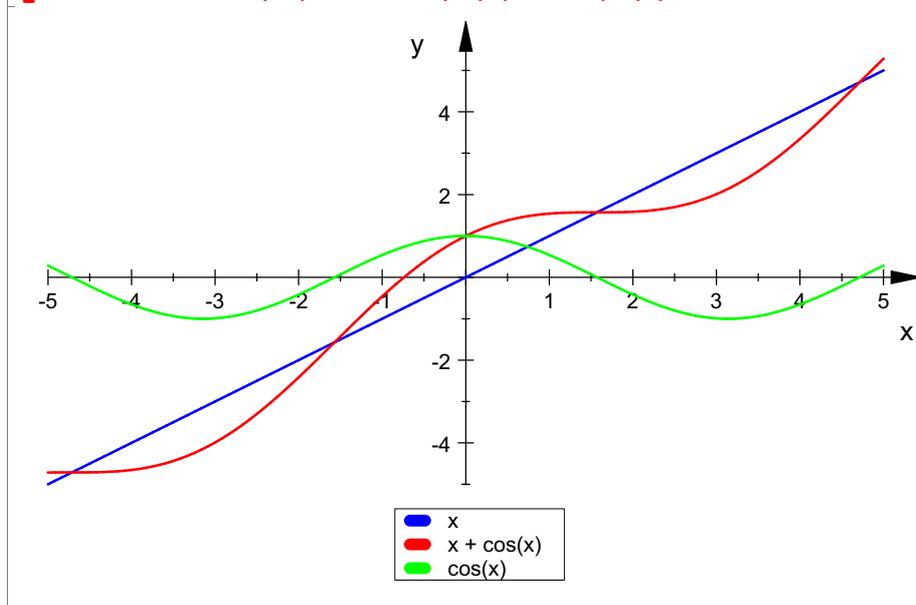
1. Aus Bausteinen aufbauen, Nullstelle berechnen
2. Funktion näherungsweise Integrieren (Fläche)
3. Funktion näherungsweise Integrieren(Rotationsvolumen)

#####

```
f:=x->x+cos(x);
```

```
x -> x + cos(x)
```

```
plotfunc2d(x,x+cos(x),cos(x))
```



Ersichtlich liegt nur eine Nullstelle vor.

Bestimmung mit MuPAD

```
xs:=numeric::solve(f(x)=0,x)[1];
```

```
-0.7390851332
```

Bestimmung mit Newtonverfahren

```
f'(x)
```

```
1 - sin(x)
```

```
newt:=x->x-f(x)/f'(x)
```

```
x -> x -  $\frac{f(x)}{f'(x)}$ 
```

```
x0:=-1.0:
```

```
[x0,f(x0),f'(x0),f(x0)/f'(x0),newt(x0)]
```

```
[-1.0, -0.4596976941, 1.841470985, -0.2496361322, -0.75036386478]
```

Hier sind die Werte, die man von Hand braucht.

Mehrfache Anwendung des Newtonverfahrens:

```
(newt @@ k)(x0) $k=1..4
```

```
(newt @@ k) (x0) $k=1..4
-0.7503638678, -0.7391128909, -0.7390851334, -0.7390851332
```

Das wird stabil, also ist der Startwert gut.

2. Funktion näherungsweise Integrieren

Beispiel: Berechnen Sie Fläche unter f im Bereich zwischen der Nullstelle und $x=3$.

```
re:=3.0:mi:=(xs+re)/2;
1.130457433
[re-xs, f(xs), f(mi), f(re)];
k1:=[(re-xs)/6, f(xs), 4*f(mi), f(re)]
[3.739085133, 0.0, 1.556703488, 2.010007503]
[0.6231808555, 0.0, 6.226813953, 2.010007503]
```

Berechnung nach der Kepler-Regel

```
kepWert:=k1[1]*(k1[2]+k1[3]+k1[4]);
5.133029442
```

zum Vergleich mit Integration

```
int(f(x), x=xs..re);
5.04160862
```

Vergleich mit einem Rechteck Breite 4 Höhe 1.6 (aus Graphen)

```
R:=4*1.6
6.4
```

Das passt.

2. Funktion näherungsweise Integrieren

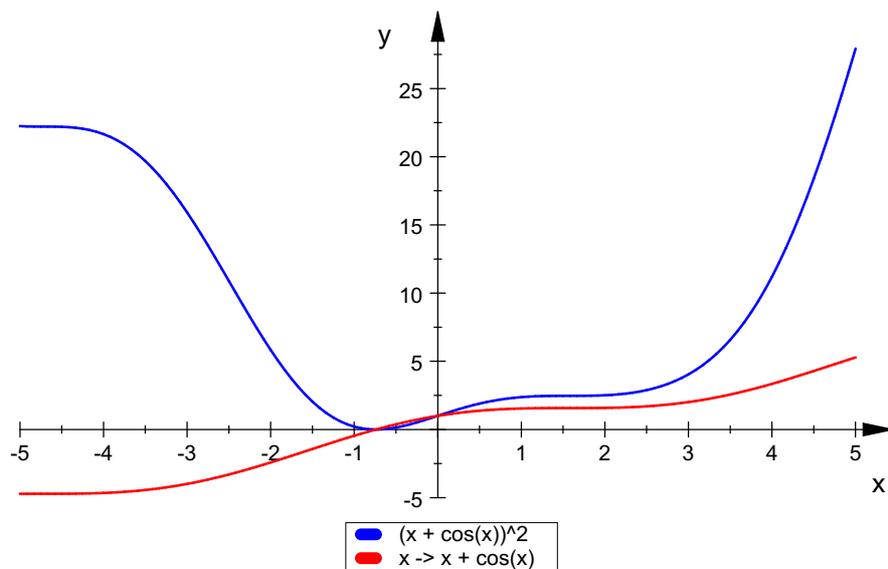
Beispiel: Berechnen Sie das Rotationsvolumen von f um die x -Achse im Bereich zwischen der Nullstelle und $x=3$.

```
f:=x->x+cos(x)
x -> x + cos(x)
```

f_q ist die Funktion, die bei einem Volumenproblem mit $f(x)=x+\cos(x)$ auftritt.

```
fq:=x->(x+cos(x))^2
x -> (x + cos(x))^2
plotfunc2d(fq(x), f)
```





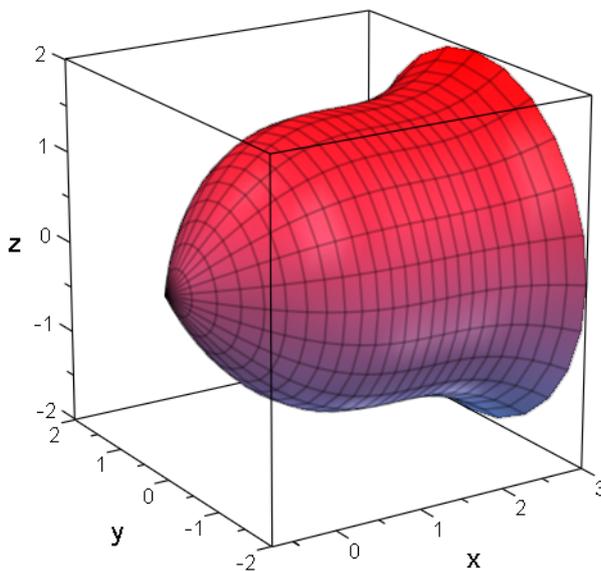
```
xs:=numeric::solve(f(x)=0,x)[1];
```

```
re:=3.0;
```

```
-0.7390851332
```

```
3.0
```

```
vase:=plot::XRotate(f(x),x=xs..re): plot(vase)
```



```
mi:=(xs+re)/2
```

```
1.130457433
```

```
fg:=plot::Function2d(f(x),x=-2..4,LineStyle=Dashed):
```

```
fqq:=plot::Function2d(fq(x),x=-2..4,LineColor=[1,0,0]):
```

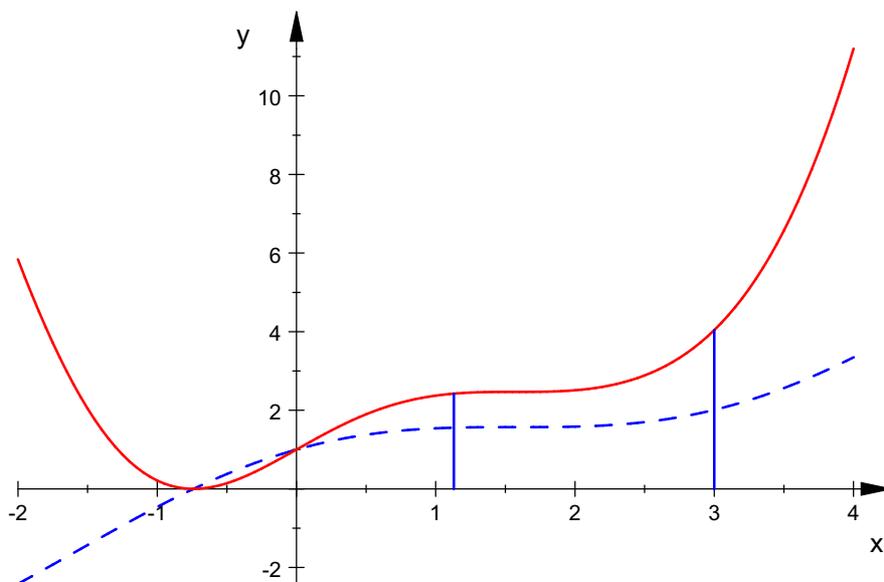
```
ende:=plot::Line2d([re,fq(re)],[re,0]):
```

```
mitte:=plot::Line2d([mi,fq(mi)],[mi,0]):
```

```
plot(fg,fqq, ende, mitte)
```

3





```
[re-xs, fq(xs),fq(mi),fq(re)];
kl:=[(re-xs)/6, fq(xs),4*fq(mi),fq(re)]
[3.739085133, 0.0, 2.423325751, 4.040130164]
[0.6231808555, 0.0, 9.693303003, 4.040130164]
```

Berechnung nach der Kepler-Regel

```
kepWert:=PI*kl[1]*(kl[2]+kl[3]+kl[4]);
float(kepWert)
8.55841263·π
26.88704625
```

zum Vergleich mit Integration

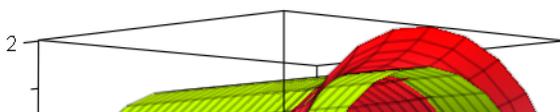
```
PI*int(fq(x),x=xs..re);
float(%)
7.576042831·π
23.8008405
```

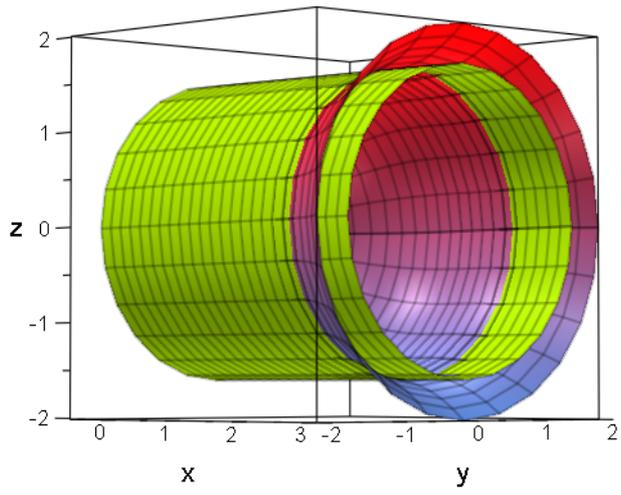
Vergleichszylinder, Radius dem Graphen entnommen

```
r:=1.6: Vz:=PI*r^2*(3-xs)
9.572057941·π
```

Das passt.

```
zy:=plot::XRotate(1.6,x=xs..3)
plot::XRotate(1.6,x=-0.7390851332..3)
plot(vase,zy)
```





#####