MWS zu Kapitel 1

Einführende Überlegungen

Wichtige MAPLE-Befehle dieses Kapitels:

PDEtools-Paket: declare
DEtools-Paket: DEplot, dfieldplot, phaseportrait
arrows (line, small, medium, ...), dirgrid, grid, stepsize
plots-Paket: animatecurve, coordplot, fieldplot, odeplot
dsolve (type=numeric, method=classical[foreuler])

1.2 Richtungsfelder

Graphische Veranschaulichung

Die Eingabe einer Differentialgleichung erfolgt in naheliegender Weise in Form einer Gleichung. Für das weitere Arbeiten mit Maple empfiehlt es sich, der Gleichung einen Namen zuzuweisen. Frühere Maple-Versionen gaben alle Ableitungen, die in einer DGL auftreten, als *partielle* Ableitungen aus. Ab Maple 8 wird die gewohnte Schreibweise $\frac{d}{dx}\,y(x)$ verwendet:

```
> restart:
  OFF: Dgl := diff(y(x),x) = f(x,y(x));
```

$$Dgl := \frac{d}{dx}\,y(x) = f(x,\,y(x))$$

Mit Hilfe des declare-Befehls aus dem PDEtools-Paket kann man Funktionen und Ableitungen in übersichtlicherer Form ausgeben. Variablen, nach denen abgeleitet wird, erscheinen als Indizes, und man kann *einer* Variablen — etwa durch die Option $prime = x$ — eine besondere Rolle geben. Ableitungen nach dieser Variablen werden dann durch Apostrophe gekennzeichnet.

Wir haben diese Anweisung (neben anderen) bereits in unsere private Initialisierungsdatei (.mapleinit bei UNIX/LINUX bzw. maple.ini bei WINDOWS) in der Form PDEtools[declare](prime=x,y(x),quiet): eingetragen. Dies hat den Vorteil, daß sie beim Hochfahren von Maple und nach jedem restart ausgeführt wird. Die zusätzliche Option quiet unterdrückt die Ausgabe von kommentierendem Text dazu. Mit Hilfe der Befehle OFF (siehe oben) und ON (siehe unten) kann man die Wirkung des *declare*-Befehls nach Belieben aus- bzw. wieder einschalten.

```
> ON: Dgl;
```

$$y' = f(x, y)$$

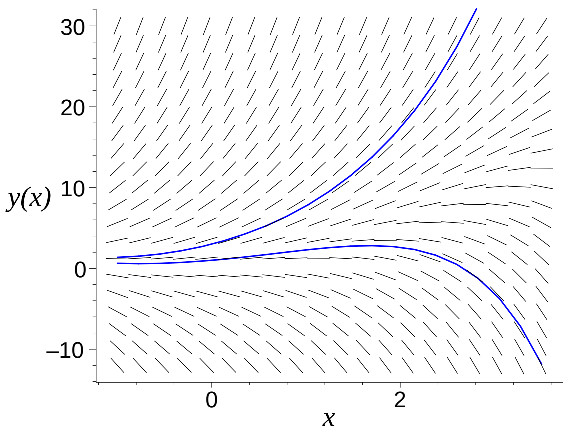
Die Veranschaulichung einer Differentialgleichung durch ein Richtungsfeld kann mit Maple auf unterschiedliche Weise realisiert werden. Wir beginnen mit dem Maple-Befehl DEplot aus dem DEtools-Paket, der aufgrund der Voreinstellung dirgrid=[20, 20] ein Gitter von 20×20 Linienelementen zeichnet. Durch eine Liste von Anfangswerten kann man einfach eine oder mehrere Lösungen in das Richtungsfeld einzeichnen:

Beispiel a)

```
> with(DEtools): with(plots): f := (x,y) -> y-x^2;
  AnfBed := [y(0) = 1], [y(0) = 3];
  # Alternativ:  AnfBed := [0,1], [0,3];
```

$$f := (x, y) \to y - x^2$$
$$\mathit{AnfBed} := [y(0) = 1],\ [y(0) = 3]$$

```
> DEplot(Dgl,y(x),x=-1..3.5,y=-12..30,[AnfBed],linecolor=blue,
    color=black,arrows=line,axes=frame,xtickmarks=3);
```



Die Option y=-12..30 kann hier mit der Liste [AnfBed] vertauscht werden:

```
> DEplot(Dgl,y(x),x=-1..3.5,[AnfBed],y=-12..30,linecolor=blue,
    color=black,arrows=line,axes=frame,xtickmarks=3):
```

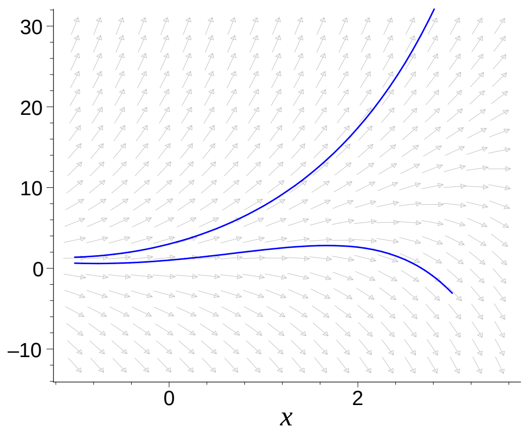
Mit phaseportrait erhält man dasselbe Bild. Hier kann die Reihenfolge der Optionen [AnfBed], y=-12..30 nicht vertauscht werden! Wir wissen nicht, ob die Maple-Entwickler sich dabei etwas gedacht haben.

```
> phaseportrait(Dgl,y(x),x=-1..3.5,[AnfBed],y=-12..30,linecolor=blue,
    color=black,arrows=line,axes=frame,xtickmarks=3):
```

Soll der Verlauf der Lösungen im Richtungsfeld durch Pfeile (an Stelle von einfachen Linienelementen) betont werden, so ist die Option arrows=line wegzulassen oder z. B. durch arrows=medium zu ersetzen.

Benötigt man nur das Richtungsfeld, so kann es auch mit dem Befehl dfieldplot aus dem DEtools-Paket gezeichnet werden. Durch Überlagerung mit zwei Lösungen der Differentialgleichung erhalten wir folgende *Animation:*

```
> dsolve({Dgl,y(0)=b},y(x)): y1 := unapply(rhs(%),x,b):
  Feld   := dfieldplot(Dgl,y(x),x=-1..3.5,y=-12..30,color=gray,
    arrows=medium,thickness=1):
  Kurven := animatecurve({y1(x,1),y1(x,3)},x=-1..3.5,color=blue,
    thickness=3,frames=50):
  display(Kurven,Feld,axes=frame,xtickmarks=3);
```



Für die Animation berechnet Maple entsprechend der Option frames=50 eine Folge von 50 Bildern. Durch Anklicken der Graphik erscheint eine Menüleiste. Ist Balloon-Help aktiviert, kann man leicht feststellen, welche Funktion die einzelnen Knöpfe haben, und damit das Ablaufen der Animation steuern. Dies stimmt mit den Tasten auf üblichen Abspielgeräten überein, ist daher ganz einfach.
Interessant ist auch folgende Animation, welche die Abhängigkeit der Lösung vom Anfangswert veranschaulicht. Wir verzichten hier auf die Ausgabe einer Momentaufnahme der Graphik.

```
> Kurven := animate(y1(x,b),x=-1..3.5,b=1..3,color=blue,thickness=3,
    frames=50):
  display(Kurven,Feld,axes=frame,xtickmarks=3);
```

Hier noch — wieder ohne Ausgabe der Graphik — eine weitere Variante:

```
> NORM := sqrt(1+f(x,y)^2):
  Feld := fieldplot([1/NORM,f(x,y)/NORM],x=-1..3.5,y=-12..30,
            color=gray,arrows=medium,thickness=1):
  display(Kurven,Feld,axes=frame,xtickmarks=3,view=-12..30):
```

Soll bei der Erzeugung eines Richtungsfeldes mittels fieldplot das voreingestellte Gitter von 20×20 Linienelementen verändert werden, so ist an Stelle von dirgrid hier grid zu verwenden.

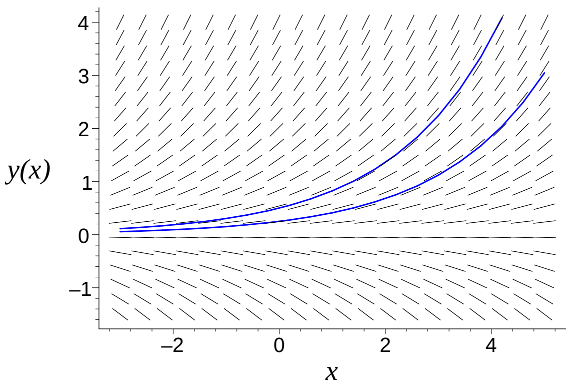
Beispiel b)

In diesem Falle hängt die Funktion f nur von y ab.

```
> f := (x,y) -> y/2;
  AnfBed := [y(0) = 1/4], [y(0) = 1/2];
```

$$f := (x,\, y) \to \frac{1}{2}\, y\,, \quad \mathit{AnfBed} := [y(0) = 1/4],\, [y(0) = 1/2]$$

```
> DEplot(Dgl,y(x),x=-3..5,y=-3/2..4,[AnfBed],linecolor=blue,color
    =black,arrows=line,axes=frame,xtickmarks=3,scaling=constrained);
```



Beispiel c)

```
> f := (x,y) -> -x/y;
  AnfBed1 := seq([y(0) = k],k=1..5);
  AnfBed2 := [y(0) = -5/2],[y(0) = -9/2];
```

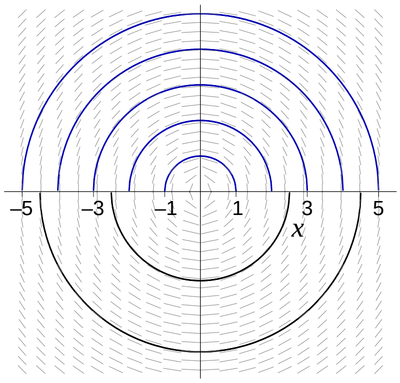
$$f := (x,\, y) \to -\frac{x}{y}$$

$$\mathit{AnfBed1} := [y(0) = 1],\, [y(0) = 2],\, [y(0) = 3],\, [y(0) = 4],\, [y(0) = 5]$$

$$\mathit{AnfBed2} := \left[y(0) = -5/2\right],\, \left[y(0) = -9/2\right]$$

Wegen der Singularität für $y = 0$ setzt sich die folgende Abbildung aus zwei Teilbildern zusammen. Ferner muß die Schrittweite mittels der Option stepsize hinreichend klein gewählt werden.

```
> DEplot(Dgl,y(x),x=-5..5,y=0.1..5,[AnfBed1],linecolor=blue,
    color=gray,arrows=line,stepsize=0.02):
  DEplot(Dgl,y(x),x=-5..5,y=-5..-0.1,[AnfBed2],linecolor=black,
    color=gray,arrows=line,stepsize=0.02):
  display(%,%%,tickmarks=[[-5,-3,-1,1,3,5],0],scaling=constrained);
```



Beispiel d)

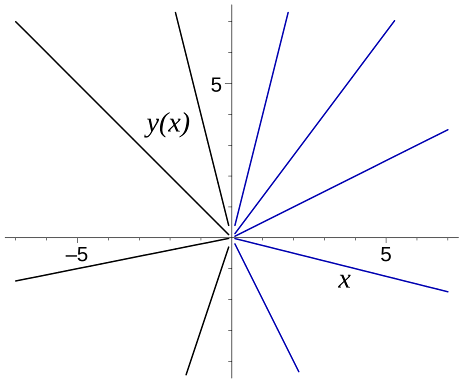
```
> f := (x,y) -> y/x;
  AnfBed1 := seq([y(1)=y0],y0=[-2,-1/4,1/2,4/3,4]);
  AnfBed2 := seq([y(-1)=y0],y0=[-3,-1/5,1,4]);
```

$$f := (x, y) \to \frac{y}{x}$$

$$AnfBed1 := [y(1) = -2],\ [y(1) = -1/4],\ [y(1) = 1/2],\ [y(1) = 4/3],\ [y(1) = 4]$$

$$AnfBed2 := [y(-1) = -3],\ [y(-1) = -1/5],\ [y(-1) = 1],\ [y(-1) = 4]$$

```
> DEplot(Dgl,y(x),x=0.1..7,y=-4..7,[AnfBed1],linecolor=blue,
    color=gray,arrows=none):
  DEplot(Dgl,y(x),x=-7..-0.1,y=-4..7,[AnfBed2],linecolor=black,
    color=gray,arrows=none):
  display(%,%%,thickness=2,tickmarks=[3,3],scaling=constrained);
```



Euler-Polygonzugverfahren

Wir betrachten die folgende einfache Anfangswertaufgabe:

```
> restart: with(plots): f := (x,y) -> x/4:
  Dgl := diff(y(x),x) = f(x,y(x));
  AnfBed := y(0) = 0;
```

$$Dgl := y' = \frac{x}{4}, \quad AnfBed := y(0) = 0$$

Obwohl deren Lösung auf der Hand liegt (Stammfunktion zu f durch $(0,0)$), berechnen wir sie symbolisch mit Hilfe des vielseitigen Maple-Befehls dsolve:

```
> dsolve({Dgl,AnfBed},y(x));
```

$$y = \frac{x^2}{8}$$

Die anschauliche Deutung einer Differentialgleichung mittels eines Richtungsfeldes legt es nahe, die exakte Lösung dieser Anfangswertaufgabe durch eine stückweise lineare Funktion zu approximieren, die man in das Richtungsfeld einpaßt: Ausgehend von einem Näherungswert Y an der Stelle X erhält man so die Näherung $Y + h \cdot f(X,Y)$ an der Stelle $X + h$.

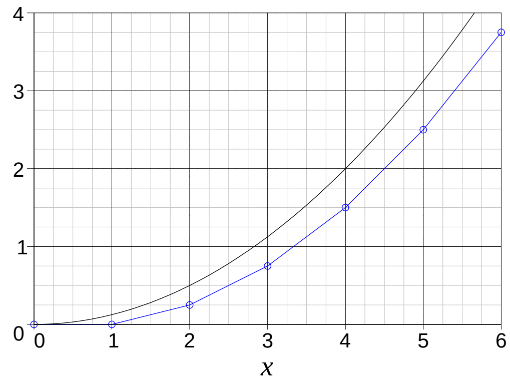
Wir berechnen die Näherungslösung mit Hilfe elementarer Maple-Befehle und unterlegen ihr in der Graphik zur Verdeutlichung ein Gitter:

```
> n := 6: Xend := 6: h := Xend/n:
  X := 0: Y := 0: S := X,Y:
  to n do
    Y := Y+h*f(X,Y);
    X := X+h;
    S := S,X,Y
  end do:
  printf("%13s %16s","x","Y(x,h)");
  printf(cat("%14.1f           %04.2f \n"$n+1),S);
```

x	$Y(x,h)$
0.0	0.00
1.0	0.00
2.0	0.25
3.0	0.75
4.0	1.50
5.0	2.50
6.0	3.75

```
> L := [[S[2*j-1],S[2*j]] $ j=1..n+1]:
  Loesung := plot(x^2/8,x=0..Xend,color=black,thickness=2):
  Punkte  := pointplot(L,symbol=circle,color=blue):
  Polygon := listplot(L,color=blue):
```

```
GitterOption := cartesian,[0..Xend,0..4],view=[0..Xend,0..4]:
Gitter := coordplot(GitterOption,grid=[7,5],color=[black $ 2]),
  coordplot(GitterOption,grid=[25,17],color=[gray $ 2]):
display(Loesung,Punkte,Polygon,Gitter,axes=frame,tickmarks=
  [[$ 0..Xend],[$ 0..4]],scaling=constrained);
```



Durch die zweifache Anwendung des Kommandos coordplot wird ein grobes (schwarzes) bzw. ein feines (graues) Gitter mit Gitterbreite 1 bzw. 1/4 erzeugt. *GitterOption* enthält die drei wichtigsten Parameter: cartesian ist der Name des gewünschten Koordinatensystems. Es folgt eine Liste von zwei Bereichen, die in der kartesischen Parameterebene das Rechteck $[0, 6] \times [0, 4]$ beschreiben, und durch view wird im Bildbereich ein Fenster für das Gitter festgelegt. Die Zahl der Gitterlinien wird mit Hilfe der Option grid gesteuert.

Leser, die mit diesem Teil des Worksheets herumexperimentieren und dabei z. B. die Schrittweite h verkleinern wollen, indem sie die Zahl n der Integrationsschritte vergrößern, werden beobachten, daß die zugehörigen Euler-Polygonzüge sich — erwartungsgemäß — der Lösung der AWA nähern.

Alternativ kann man die Lösung mit dsolve unter Verwendung der Optionen type=numeric und method=classical[foreuler] näherungsweise berechnen. Mit Hilfe des Befehls odeplot aus dem plots-Paket kann man sie zeichnen:

```
> p := dsolve({Dgl,AnfBed},y(x),type=numeric,
        method=classical[foreuler],stepsize=h,
        output=array([evalf(k*h) $ k=0..n])):
  eval(p);
```

$$[x,\, y]$$
$$\begin{bmatrix} 0 & 0.00 \\ 1 & 0.00 \\ 2 & 0.25 \\ 3 & 0.75 \\ 4 & 1.50 \\ 5 & 2.50 \\ 6 & 3.75 \end{bmatrix}$$

```
> Punkte  := odeplot(p,style=point,symbol=circle,color=blue):
  Polygon := odeplot(p,color=blue):
  display(Loesung,Punkte,Polygon,Gitter,axes=frame,
          tickmarks=[[$ 0..Xend],[$ 0..4]],scaling=constrained):
```

Da die Graphik nahezu identisch ist mit der vorangehenden Abbildung, verzichten wir hier auf eine Wiedergabe.