1. Anfangswertprobleme 1. Ordnung

1.1 Grundlagen

- Zu einem Anfangswertproblem 1. Ordnung gehören folgende Daten:
  - Eine Differenzialgleichung 1. Ordnung: $\dot{x}(t) = f\left[x(t), t\right]$
  - Die Anfangsbedingung: $x(0) = x_0$
  - Das zu untersuchende Zeitintervall: $t \in \left[0, t_E\right]$

1.1 Grundlagen

- Beispiel:
  - Für den Fall mit Luftwiderstand lautet die Bewegungsgleichung:

$$m\dot{v} = m g - C_W v^2 \quad \rightarrow \quad \dot{v} = g - \frac{C_W}{m} v^2$$

  - Mit $x = v$ gilt: $\dot{v} = f\left[v(t), t\right] = g - \frac{C_W}{m} v^2$
  - Um die Geschwindigkeit im Intervall $\left[0, \; t_E\right]$ berechnen zu können, muss die Anfangsgeschwindigkeit v_0 zum Zeitpunkt $t = 0$ gegeben sein.

1.1 Grundlagen

- Grundidee:
  - Bei allen numerischen Verfahren wird der Wert der gesuchten Funktion für diskrete Zeitpunkte ermittelt.
  - Dazu wird das Zeitintervall in äquidistante Zeitschritte *Δt* unterteilt:

$$\Delta t = \frac{t_n - t_0}{n} = \frac{t_E}{n}, \quad t_i = t_0 + i\,\Delta t, \quad i = 0, \dots, n$$

1.1 Grundlagen

- Abkürzungen: $x(t_i) = x_i\,,\quad \dot{x}(t_i) = \dot{x}_i$

- Integration der Differenzialgleichung von t_i bis t_{i+1} ergibt:

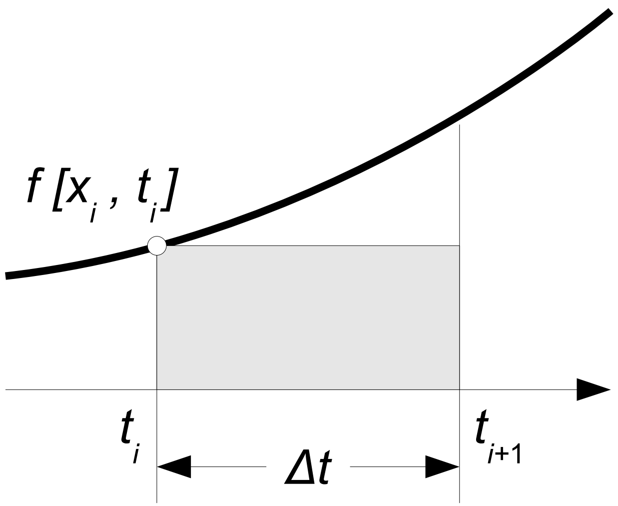
$$x_{i+1} - x_i = \int_{t_i}^{t_{i+1}} f[x(t), t]\,dt \quad \rightarrow \quad \boxed{x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} f[x(t), t]\,dt}$$

- Diese Gleichung ist der Ausgangspunkt aller numerischen Verfahren.
- Dabei wird das Integral näherungsweise berechnet.

1.2 Euler-Vorwärts-Verfahren

- Idee:
  - Beim Euler-Vorwärts-Verfahren wird das Integral approximiert durch

$$\int_{t_i}^{t_{i+1}} f\left[x(t), t\right] dt \approx f\left[x_i, t_i\right] \Delta t$$



1.2 Euler-Vorwärts-Verfahren

- Algorithmus:
  - Die Approximation des Integrals führt unmittelbar auf folgende Berechnungsvorschrift:

$$\boxed{x_{i+1} = x_i + f\left[x_i, t_i\right]\Delta t}$$

  - Der Wert zum Zeitpunkt t_{i+1} lässt sich durch eine einfache Funktionsauswertung zum Zeitpunkt t_i berechnen.
  - Derartige Verfahren werden als explizite Verfahren bezeichnet.

1.2 Euler-Vorwärts-Verfahren

- Beispiel: Fall mit Luftwiderstand
  - Funktion: $f[t, v(t)] = g - \frac{C_W}{m} v^2$
  - Daten:
    - $m = 5\,kg$, $C_W = 0{,}0162\,kg/m$
    - $t_0 = 0\,s$, $t_E = 20\,s$
    - $v_0 = 0\,m/s$

1.2 Euler-Vorwärts-Verfahren

- Programmierung in Octave:

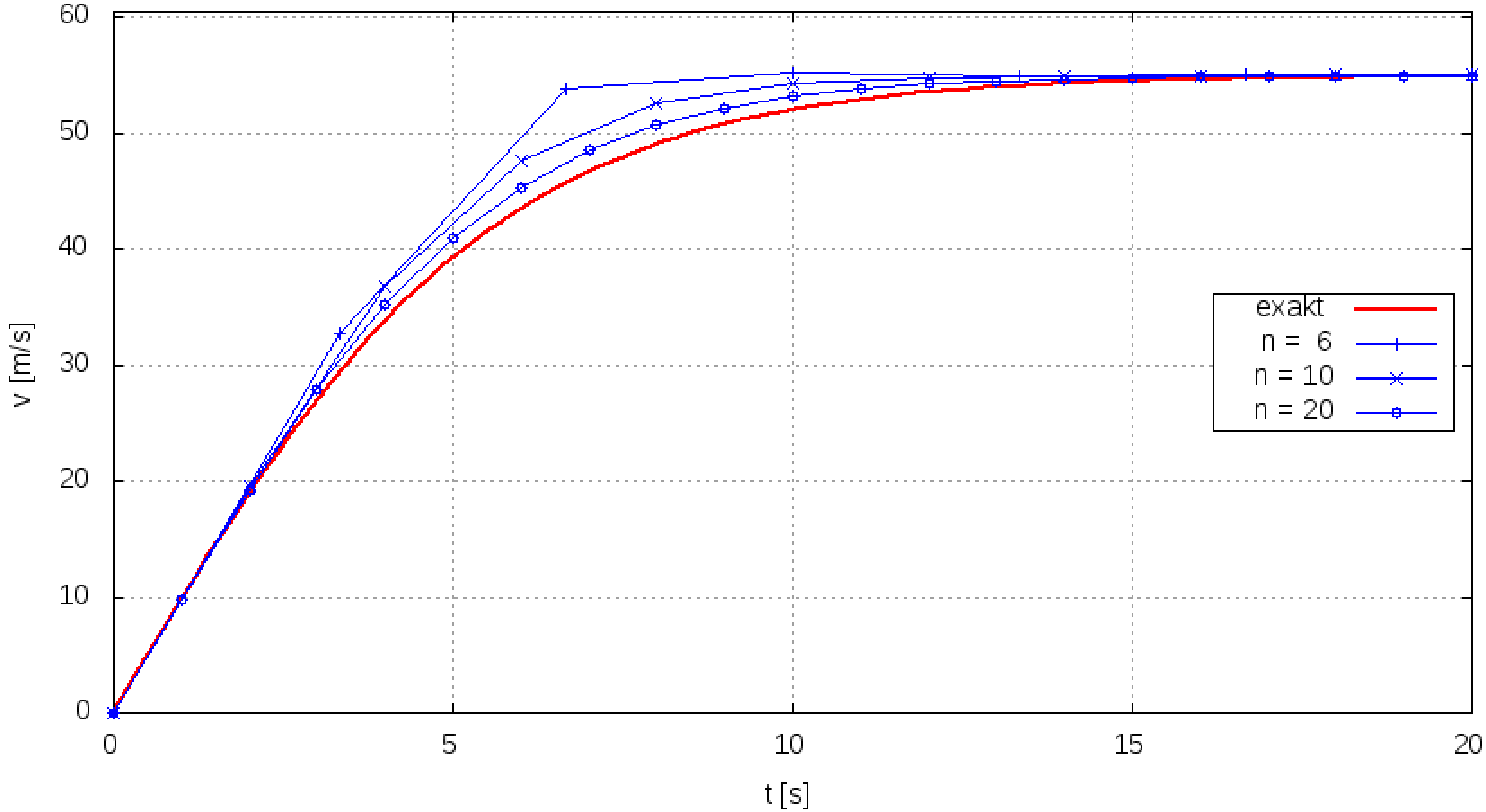
```
# Daten

  m  = 5     ;  % Masse in kg
  CW = 0.0162;  % Widerstandskonstante in kg/m
  v0 = 0     ;  % Anfangsgeschwindigkeit in m/s

  t0 = 0     ;  % Anfangszeit in s
  tE = 20    ;  % Endzeit in s

  n  = 20    ;  % Zeitintervalle

  global g  = 9.81  ;  % Erdbeschleunigung in  m/s^2
```

1.2 Euler-Vorwärts-Verfahren

```
# Definition der Funktion

  global c = CW / m;

  function y = f(x, t)
     global g;  global c;
     y = g - c * x^2;
  end

# Euler-Vorwärts-Verfahren

  v = zeros(1, n + 1);
  t = linspace(t0, tE, n + 1);
  dt = (tE - t0) / n;

  v(1) = v0;

  for i = 1 : n
      v(i + 1) = v(i) + f(v(i), t(i)) * dt;
  end
```

1.2 Euler-Vorwärts-Verfahren

- Ergebnis:



1.2 Euler-Vorwärts-Verfahren

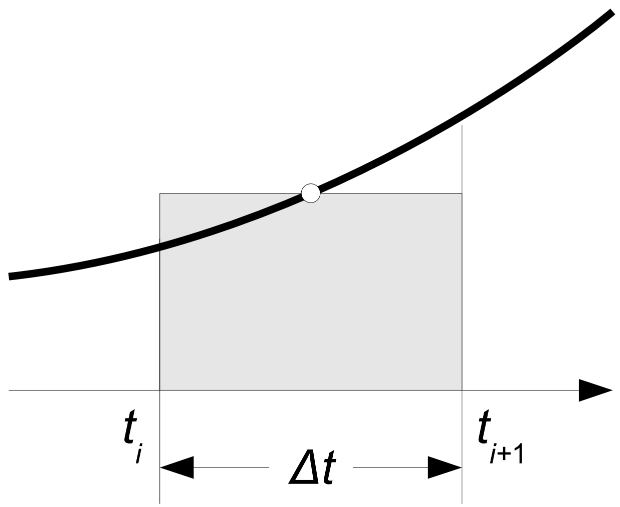
- Bewertung:
  - Das Verfahren ist einfach zu programmieren.
  - Der Rechenaufwand für einen Zeitschritt ist gering.
  - Das Verfahren ist genau von 1. Ordnung.
  - Um eine hinreichend genaue Näherung zu erhalten, muss der Zeitschritt klein sein.
  - Das Verfahren wird hauptsächlich zur Berechnung kurzzeitiger Vorgänge (Stoßvorgänge, Crash, Explosionen) verwendet.

1.3 Runge-Kutta-Verfahren

- Idee:
  - Beim zweistufigen Runge-Kutta-Verfahren wird das Integral mit dem Funktionswert in der Mitte des Intervalls approximiert:

$$\int_{t_i}^{t_{i+1}} f\left[x(t), t\right] dt$$

$$\approx f\left[x_{i+1/2}, t_i + \frac{\Delta t}{2}\right] \Delta t$$



1.3 Runge-Kutta-Verfahren

- Der Funktionswert $x_{i+1/2}$ wird durch einen Euler-Schritt berechnet:

$$x_{i+1/2} = x_i + f\left[x_i, t_i\right] \frac{\Delta t}{2}$$

- Das klassische Verfahren ist das vierstufige Runge-Kutta-Verfahren.
- Dabei fallen vier Euler-Schritte an.

1.3 Runge-Kutta-Verfahren

- Algorithmen:
  - Zweistufiges Runge-Kutta-Verfahren:

$$k_1 = f\left[x_i, t_i\right]$$

$$k_2 = f\left[x_i + k_1 \frac{\Delta t}{2}, t_i + \frac{\Delta t}{2}\right]$$

$$x_{i+1} = x_i + k_2 \Delta t$$

1.3 Runge-Kutta-Verfahren

- Vierstufiges Runge-Kutta-Verfahren:

$$k_1 = f\left[x_i, t_i\right], \qquad k_2 = f\left[x_i + k_1 \frac{\Delta t}{2}, t_i + \frac{\Delta t}{2}\right]$$

$$k_3 = f\left[x_i + k_2 \frac{\Delta t}{2}, t_i + \frac{\Delta t}{2}\right], \qquad k_4 = f\left[x_i + k_3 \Delta t, t_i + \Delta t\right]$$

$$k = \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$

$$x_{i+1} = x_i + k\,\Delta t$$

1.3 Runge-Kutta-Verfahren

- Beispiel: Fall mit Luftwiderstand
  - Die Gleichung wird mit dem vierstufigen Runge-Kutta-Verfahren gelöst.
  - Programmierung in Octave:

```
# Daten

  m  = 5     ;  % Masse in kg
  CW = 0.0162;  % Widerstandskonstante in kg/m
  v0 = 0     ;  % Anfangsgeschwindigkeit in m/s

  t0 = 0     ;  % Anfangszeit in s
  tE = 20    ;  % Endzeit in s

  n  = 6     ;  % Zeitintervalle

  global g  = 9.81  ;  % Erdbeschleunigung in  m/s^2
```

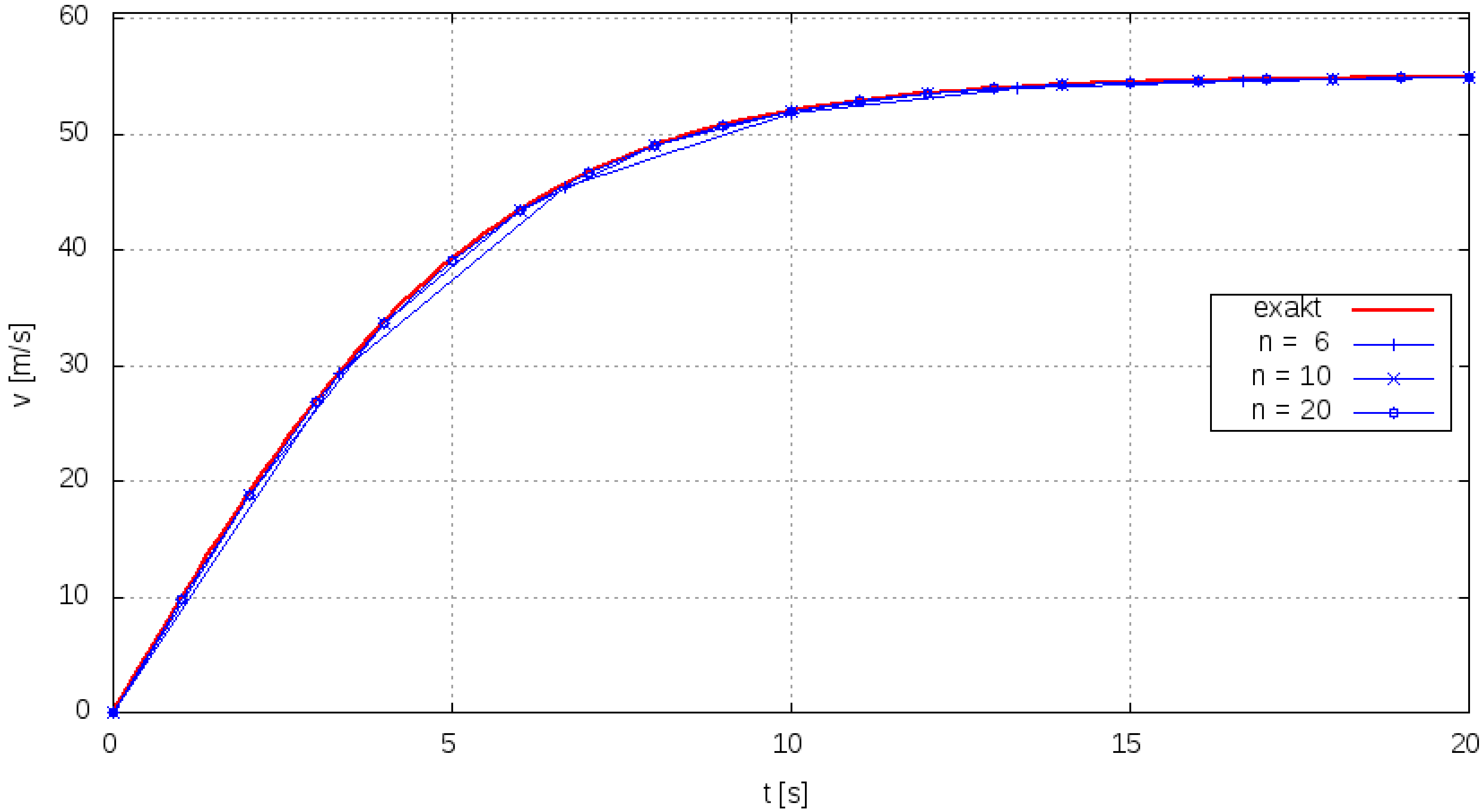
1.3 Runge-Kutta-Verfahren

```
# Definition der Funktion

  global c = CW / m;

  function y = f(x, t)
     global g; global c;
     y = g - c * x^2;
  end
```

```
# Runge-Kutta-Verfahren

  v = zeros(1, n + 1);
  t = linspace(t0, tE, n + 1);
  dt = (tE - t0) / n;

  v(1) = v0;

  for i = 1 : n
      k1 = f(v(i), t(i));
      k2 = f(v(i) + k1 * dt / 2, t(i) + dt / 2);
      k3 = f(v(i) + k2 * dt / 2, t(i) + dt / 2);
      k4 = f(v(i) + k3 * dt, t(i) + dt);
      k  = (k1 + 2 * k2 + 2 * k3 + k4) / 6;
      v(i + 1) = v(i) + k * dt;
  end
```

1.3 Runge-Kutta-Verfahren

- Ergebnis:



1.3 Runge-Kutta-Verfahren

- Bewertung:
  - Die Runge-Kutta-Verfahren sind wie das Vorwärts-Euler-Verfahren explizite Verfahren.
  - Das zweistufige Runge-Kutta-Verfahren ist genau von 2. Ordnung und das vierstufige Verfahren ist genau von 4. Ordnung.
  - Voraussetzung für die Genauigkeit ist, dass die Funktion 3 bzw. 5 mal stetig nach der Zeit differenzierbar ist.

1.4 Stabilität

- Definition:
  - Ein numerisches Verfahren heißt stabil, wenn Fehler, die im Laufe der Berechnung anfallen, nicht anwachsen.
  - Dann bleibt die numerische Lösung beschränkt, wenn die exakte Lösung beschränkt ist.
- Stabilität des Euler-Vorwärts-Verfahrens:
  - Betrachtet wird die Differenzialgleichung $\dot{x}(t) = -c\,x(t)$.
  - Die Rechenvorschrift für das Euler-Vorwärts-Verfahren lautet:

$$x_{i+1} = x_i - c\,x_i\,\Delta t = (1 - c\,\Delta t)\,x_i$$

1.4 Stabilität

- In diesem Fall lässt sich der Wert x_{i+1} auch direkt aus dem Anfangswert berechnen: $x_{i+1}=(1-c\Delta t)^i x_0$
- Für $|1-c\Delta t|<1$ strebt diese Folge gegen null. Das entspricht dem asymptotischen Verhalten der exakten Lösung
$$x(t)=x_0\exp(-ct)$$
- Für $|1-c\Delta t|>1$ wächst die Folge über alle Grenzen. Das Verfahren ist instabil.
- Die Stabilität des Euler-Vorwärts-Verfahrens hängt also von der Größe des Zeitschrittes ab.
- Für das angegebene Beispiel ist das Verfahren stabil für
$$\Delta t<2/c$$

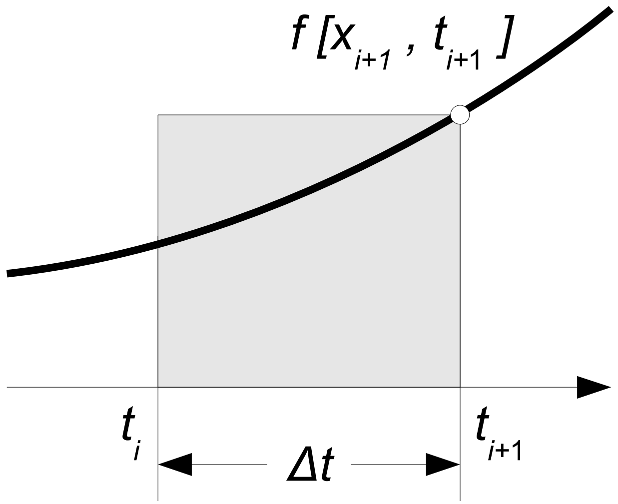
1.4 Stabilität

- Bei allen expliziten Verfahren darf der Zeitschritt einen bestimmten Wert nicht überschreiten, damit das Verfahren stabil ist.

1.5 Euler-Rückwärts-Verfahren

- Idee:
  - Beim Euler-Rückwärts-Verfahren wird das Integral approximiert durch

$$\int_{t_i}^{t_{i+1}} f\left[x(t), t\right] dt \approx f\left[x_{i+1}, t_{i+1}\right] \Delta t$$



1.5 Euler-Rückwärts-Verfahren

- Algorithmus:
  - Die Approximation des Integrals führt auf folgende Berechnungsvorschrift:

$$x_{i+1} = x_i + f\left[x_{i+1}, t_{i+1}\right] \Delta t$$

  - Zur Berechnung des Wertes zum Zeitpunkt t_{i+1} muss eine im Allgemeinen nichtlineare Gleichung gelöst werden.
  - Die nichtlineare Gleichung kann z.B. mit dem Newton-Raphson-Verfahren gelöst werden.
  - Verfahren, bei denen in jedem Zeitschritt eine Gleichung gelöst werden muss, werden als implizite Verfahren bezeichnet.

1.5 Euler-Rückwärts-Verfahren

- Beispiel: Fall mit Luftwiderstand
  - Die Daten sind die gleichen wie beim Euler-Vorwärts-Verfahren.
  - Die Gleichung für einen Zeitschritt lautet:

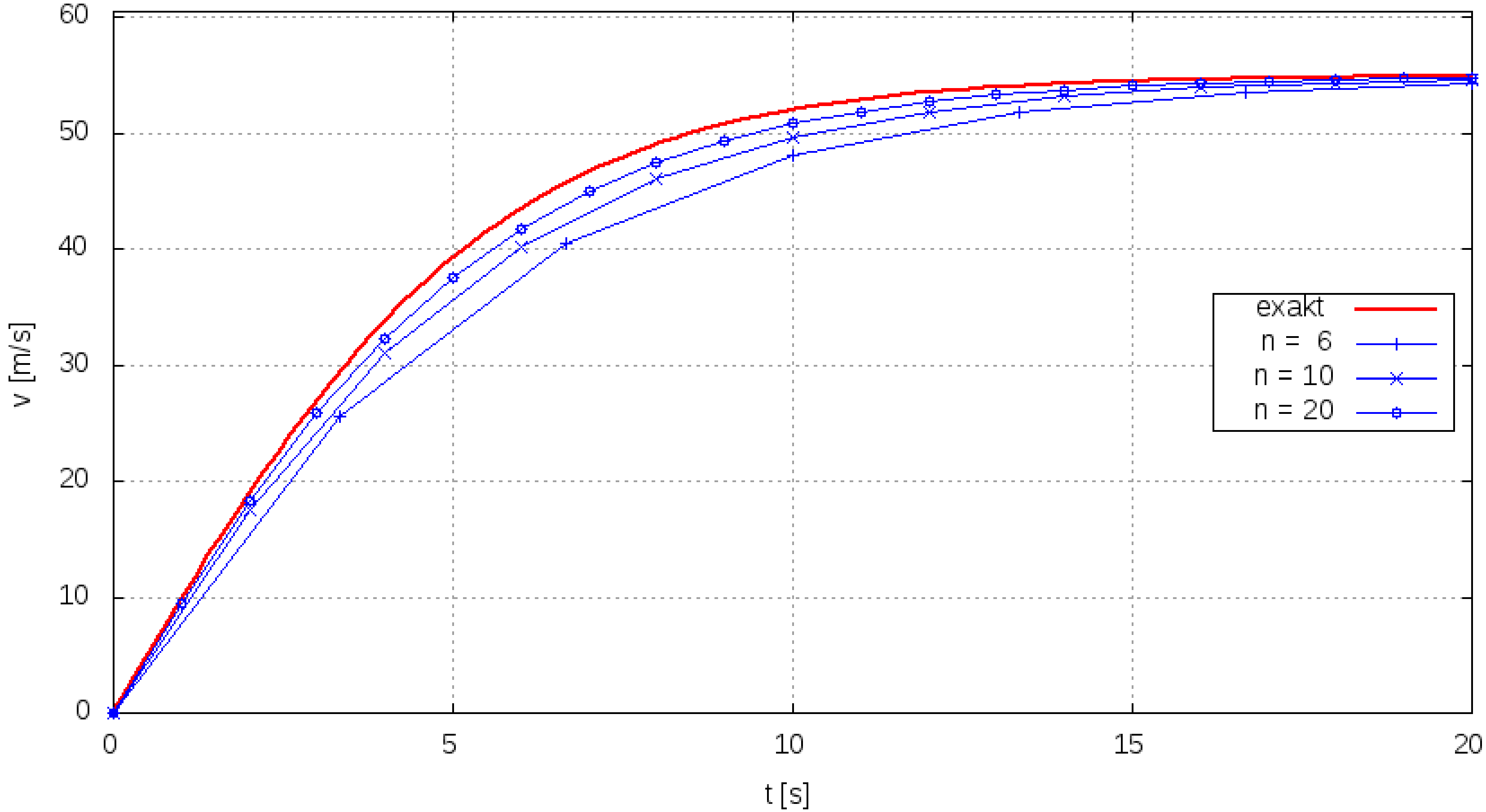
$$v_{i+1} = v_i + \left(g - \frac{C_W}{m} v_{i+1}^2 \right) \Delta t$$

  - Sie hat die Lösung

$$v_{i+1} = -\frac{m}{2 C_W \Delta t} + \sqrt{\left(\frac{m}{2 C_W \Delta t} \right)^2 + \frac{m}{C_W \Delta t} \left(v_i + g \Delta t \right)}$$

1.5 Euler-Rückwärts-Verfahren

- Ergebnis:



1.5 Euler-Rückwärts-Verfahren

- Stabilität:
  - Für die Differentialgleichung $\dot{x}(t) = -c\,x(t)$

    lautet die Rechenvorschrift: $x_{i+1} = x_i - c\,x_{i+1}\,\Delta t$
  - Daraus folgt: $x_{i+1} = \dfrac{1}{1 + c\,\Delta t}\,x_i$
  - Der Faktor, mit dem der Wert x_i multipliziert wird, ist unabhängig vom Zeitschritt immer kleiner als 1.
  - Das Euler-Rückwärts-Verfahren ist für alle Zeitschritte stabil.

1.5 Euler-Rückwärts-Verfahren

- Implizite Verfahren sind in der Regel stabiler als explizite Verfahren, d.h. sie bleiben auch für größere Zeitschritte stabil.
- Die Genauigkeit nimmt jedoch genauso wie bei expliziten Verfahren mit größeren Zeitschritten ab.

1.5 Euler-Rückwärts-Verfahren

- Bewertung:
  - Das Verfahren ist aufwändiger zu programmieren als das Euler-Vorwärts-Verfahren.
  - Bei nichtlinearen Gleichungen ist der Rechenaufwand für einen Zeitschritt deutlich größer als beim Euler-Vorwärts-Verfahren, da eine nichtlineare Gleichung gelöst werden muss.
  - Das Verfahren bleibt auch für größere Zeitschritte stabil.
  - Das Verfahren ist genau von 1. Ordnung.
  - Um eine hinreichend genaue Näherung zu erhalten, muss der Zeitschritt klein sein.

1.6 Differenzialgleichungssysteme

- Grundlagen:
  - Ein System von *m* Differenzialgleichungen 1. Ordnung hat die allgemeine Form

$$\begin{aligned} \dot{x}_1 &= f_1\left[x_1(t), \ldots, x_m(t), t\right] \\ &\vdots \\ \dot{x}_m &= f_m\left[x_1(t), \ldots, x_m(t), t\right] \end{aligned}$$

  - Dazu kommen die Anfangsbedingungen

$$x_1(0) = x_{01}, \quad \ldots, \quad x_m(0) = x_{0m}$$

1.6 Differenzialgleichungssysteme

- Mit den Matrizen

$$[\boldsymbol{x}(t)] = \begin{bmatrix} x_1(t) \\ \vdots \\ x_m(t) \end{bmatrix} \quad \text{und} \quad \left[\boldsymbol{f}\left[[\boldsymbol{x}(t)],t\right]\right] = \begin{bmatrix} f_1\left[[\boldsymbol{x}(t)],t\right] \\ \vdots \\ f_m\left[[\boldsymbol{x}(t)],t\right] \end{bmatrix}$$

lässt sich das Anfangswertproblem in der Form

$$[\dot{\boldsymbol{x}}(t)] = \left[\boldsymbol{f}\left[[\boldsymbol{x}(t)],t\right]\right], \quad [\boldsymbol{x}(0)] = [\boldsymbol{x}_0]$$

schreiben.

1.6 Differenzialgleichungssysteme

- Lösungsverfahren:
  - Die vorgestellten Lösungsverfahren lassen sich unmittelbar auf Systeme von Differenzialgleichungen übertragen, indem die skalaren Größen durch die entsprechenden Matrizen ersetzt werden.
  - Euler-Vorwärts-Verfahren: $\left[\boldsymbol{x}_{i+1}\right]=\left[\boldsymbol{x}_i\right]+\left[\boldsymbol{f}\left[\left[\boldsymbol{x}_i\right], t_i\right]\right]\Delta t$
  - Euler-Rückwärts-Verfahren: $\left[\boldsymbol{x}_{i+1}\right]=\left[\boldsymbol{x}_i\right]+\left[\boldsymbol{f}\left[\left[\boldsymbol{x}_{i+1}\right], t_{i+1}\right]\right]\Delta t$
    - Für jeden Zeitschritt muss ein im Allgemeinen nichtlineares Gleichungssystem gelöst werden.

1.6 Differenzialgleichungssysteme

- Vierstufiges Runge-Kutta-Verfahren:

$$[\boldsymbol{k}_1] = \left[\boldsymbol{f} \left[[\boldsymbol{x}_i], t_i \right] \right]$$

$$[\boldsymbol{k}_2] = \left[\boldsymbol{f} \left[[\boldsymbol{x}_i] + [\boldsymbol{k}_1] \Delta t^2, t_i + \frac{\Delta t}{2} \right] \right]$$

$$[\boldsymbol{k}_3] = \left[\boldsymbol{f} \left[[\boldsymbol{x}_i] + [\boldsymbol{k}_2] \Delta t^2, t_i + \frac{\Delta t}{2} \right] \right]$$

$$[\boldsymbol{k}_4] = \left[\boldsymbol{f} \left[[\boldsymbol{x}_i] + [\boldsymbol{k}_3] \Delta t, t_i + \Delta t \right] \right]$$

$$[\boldsymbol{k}] = \frac{1}{6} \left([\boldsymbol{k}_1] + 2 [\boldsymbol{k}_2] + 2 [\boldsymbol{k}_3] + [\boldsymbol{k}_4] \right)$$

$$[\boldsymbol{x}_{i+1}] = [\boldsymbol{x}_i] + [\boldsymbol{k}] \Delta t$$

1.6 Differenzialgleichungssysteme

- Beispiel: Ballistischer Wurf
  - Der ballistische Wurf ist ein schiefer Wurf, bei dem der Luftwiderstand berücksichtigt wird.
  - Die Bewegungsgleichungen lauten

$$
\begin{aligned}
m\,\dot{u} &= -C_W\,u\sqrt{u^2+w^2}\\
m\,\dot{w} &= -m\,g-C_W\,w\sqrt{u^2+w^2}\\
\dot{x} &= u\\
\dot{z} &= w
\end{aligned}
$$

1.6 Differenzialgleichungssysteme

- In Standardform lautet das System von Differenzialgleichungen:

$$[\dot{\boldsymbol{x}}]=\begin{bmatrix}\dot{u}\\ \dot{w}\\ \dot{x}\\ \dot{z}\end{bmatrix}=\begin{bmatrix}-\dfrac{C_W}{m}u\sqrt{u^2+w^2}\\ -g-\dfrac{C_W}{m}w\sqrt{u^2+w^2}\\ u\\ w\end{bmatrix}=\left[\boldsymbol{f}\left[[\boldsymbol{x}],t\right]\right]$$

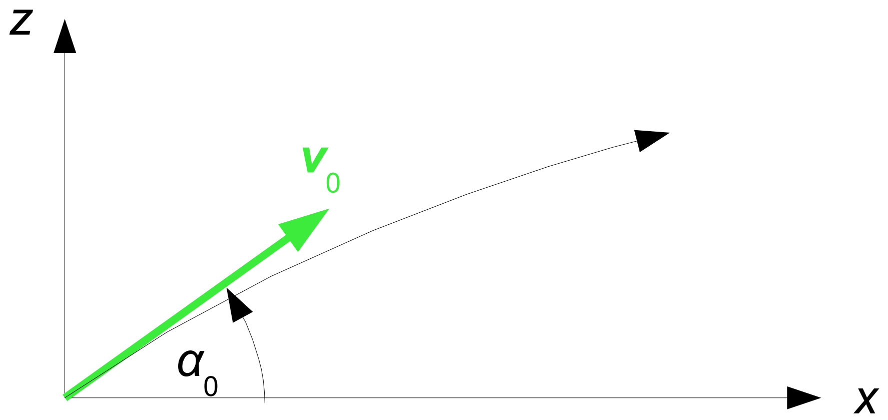
1.6 Differenzialgleichungssysteme

- Anfangsbedingungen:

$$x(0)=x_0,\quad z(0)=z_0$$

$$u(0)=u_0=v_0\cos(\alpha_0)$$

$$w(0)=w_0=v_0\sin(\alpha_0)$$



- Zahlenwerte:
  - m = 1kg, C_W = 0,04kg/m
  - v_0 = 15m/s, α_0 = 50°
  - x_0 = 0m, z_0 = 0m

- Die Flugbahn wird mit dem vierstufigen Runge-Kutta-Verfahren berechnet und mit der Bahn ohne Luftwiderstand verglichen.

1.6 Differenzialgleichungssysteme

- Ergebnis: