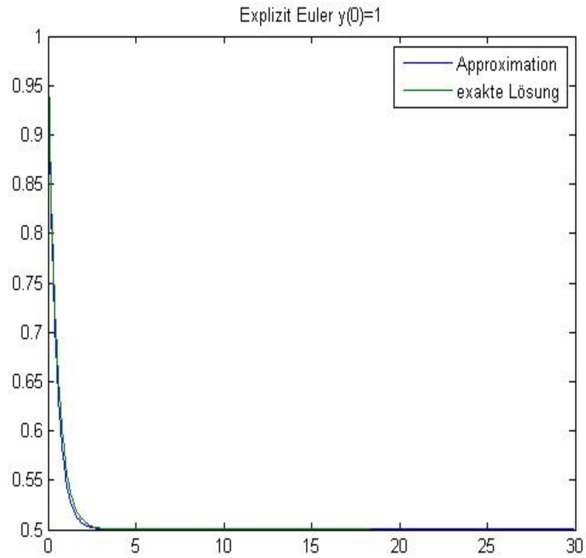
Fabian Meyer; Ergebnispin: 8104

Labor: Modelle für virtuelle Realität

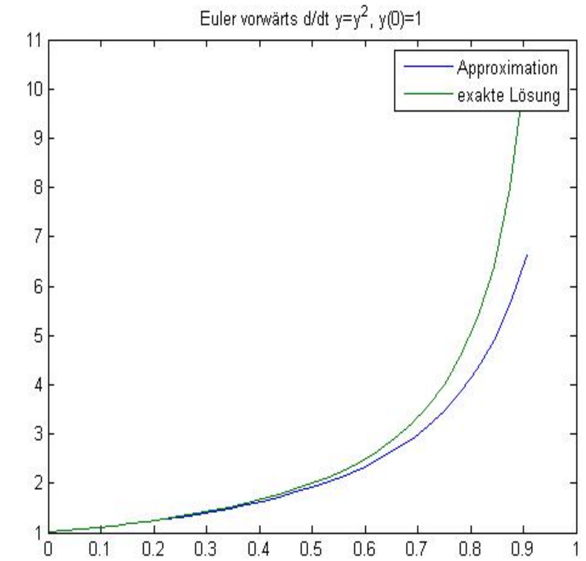
Der Quellcode wurde in Java Processing geschrieben, die Lösungsvektoren wurden dann mit Matlab geplottet. Die Anfangswerte habe ich mir selbst vorgegeben, ebenso das Intervall und die Diskretisierung. Die exakte Lösung wurde mit Wolfram Alpha bestimmt.

Aufgabe 1.1 (Explizite Integration)

- $f'(x) + 2f(x) = 1$. Wir wählen $I = [0, 30]$, $h = \frac{1}{100}$ und $f(0) = 1$. Die exakte Lösung ist $f(x) = \frac{1}{2}(e^{-2x} + 1)$. Wie zu erkennen ist, ist die numerische Lösung schon ziemlich gut.

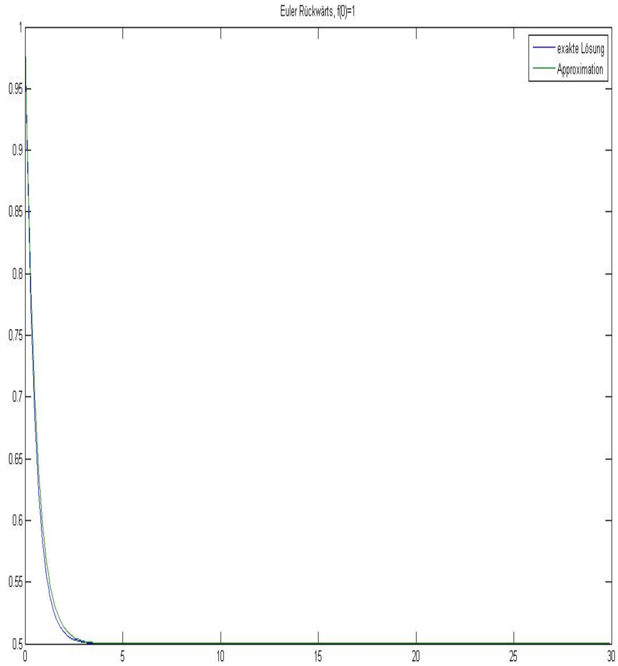


- $y' = y^2$. $I = [0, 1]$, $h = \frac{1}{32}$ und $y(0) = 1$. Wir berechnen nur 30 Schritte, weil die Approximierende im Grenzwert von links gegen die 1 explodiert. Die exakte Lösung ist $y(x) = \frac{1}{1-x}$. Die Lösung approximiert die exakte Lösung nur links der Polstelle.



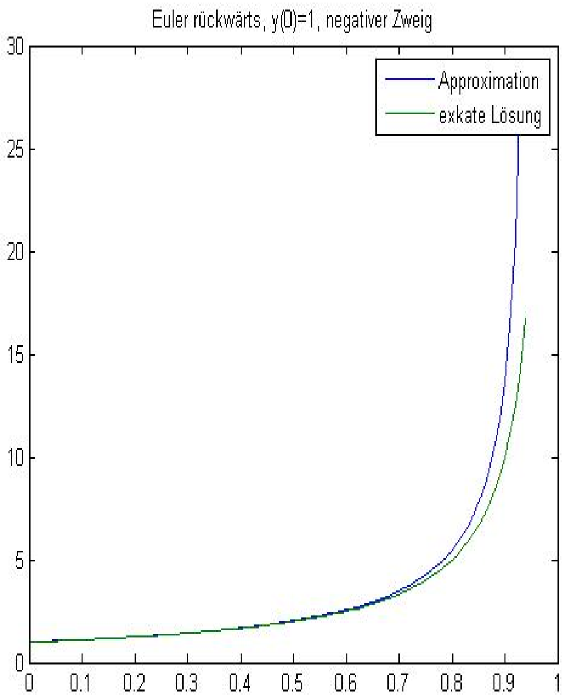
Aufgabe 1.2 (Implizite Integration)

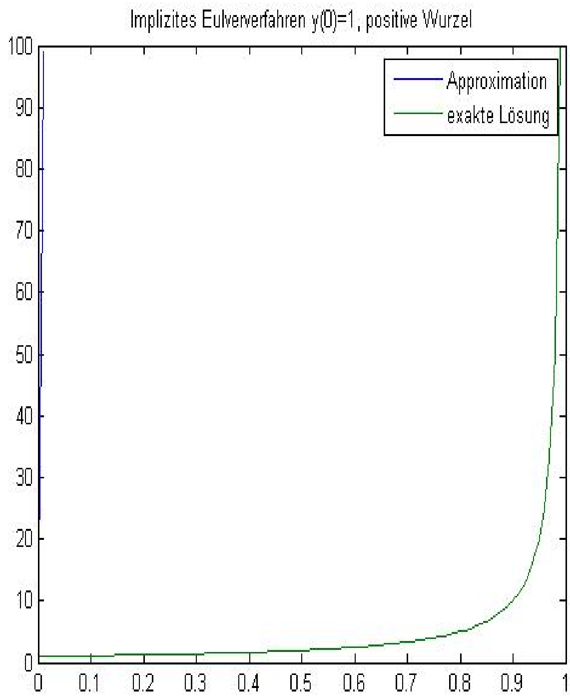
- $f'(x) + 2f(x) = 1$. $I = [0, 30], h = \frac{1}{10}$ und $f(0) = 1$. Die exakte Lösung wird relativ gut durch



  das implizite Eulerverfahren approximiert.

- $y' = y^2$ mit $y(0) = 1$, $h = 0.01, I = [0, 1]$, 96 Schritte mit dem impliziten Eulerverfahren. Das implizite Euler-Verfahren schreibt sich $y_{n+1} = y_n + hf(x_{x+1}, y_{n+1})$ und nach umstellen erhalten wir $hy_{n+1}^2 - y_{n+1} + y_n = 0$.
  $\Rightarrow y_{n+1} = \frac{1 \pm \sqrt{1-4hy_n}}{2h}$ Hier ist zu erkennen, dass nur die Lösung mit dem negativen Vorzeichen





die Funktion halbwegs gut approximiert. Die Lösung mit dem positiven Vorzeichen explodiert schon vorher. Auch hier bekommen wir rechts der Polstelle keine Approximation, da die Euler-Lösung gegen unendlich verschwindet.

Aufgabe 1.3 (Mehrdimensionale Differentialgleichungen)

a) Die Idee besteht darin, dass explizite Eulerverfahren komponentenweise anzuwenden.

- Wir schreiben $x(t) := f_x(t)$ und $y(t) := f_y(t)$ für die Komponenten von $f(t)$.
  Nun schreibt sich das DGL-System zu

$$\begin{pmatrix} x'(t) \\ y'(t) \end{pmatrix} = \begin{pmatrix} -x(t)y(t) \\ -y(t) \end{pmatrix}$$

  wobei wir wieder beliebige Aw $x(x_0) = \tilde{x},\ y(x_0) = \tilde{y}$ vorgeben. Wir wenden nun das explizite Euler-Verfahren auf beide Komponenten an und erhalten eine 2D-Approximation der exakten Lösung.

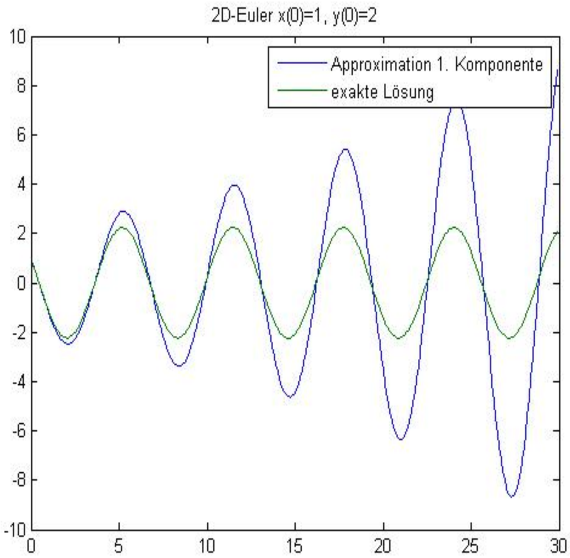
- Wir verfahren analog wie zuvor und erhalten das System

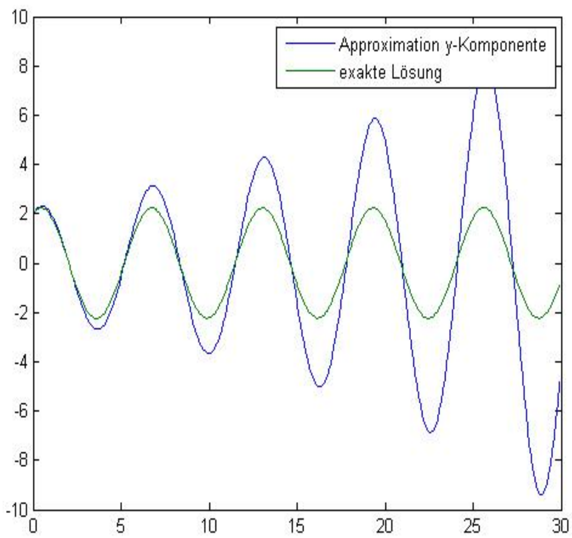
$$\begin{pmatrix} x'(t) \\ y'(t) \end{pmatrix} = \begin{pmatrix} -y(t) \\ x(t) \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$$

b) Das System kann mit der Theorie zu linearen, homogenen, DGL-Systemen gelöst werden (Eigenwerte der System Matrix A berechnen).
Die Lösungen sind

$$x(t) = c_1 cos(t) - c_2 sin(t)$$
$$y(t) = c_1 sin(t) + c_2 cos(t)$$

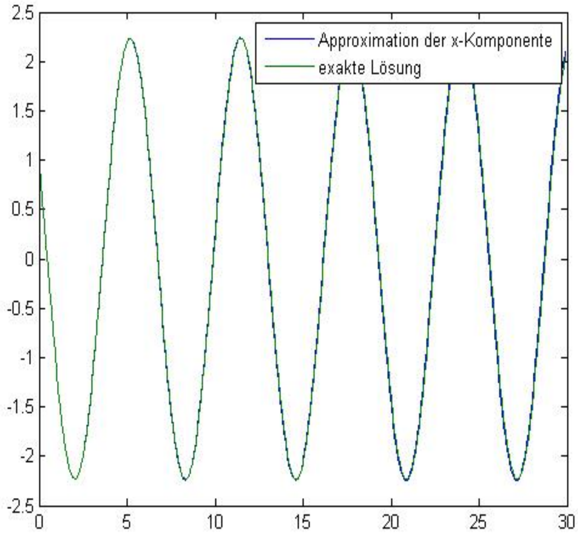
Geben wir uns die Anfangswerte $x(0) = 1,\ y(0) = 2$ vor, liefert das explizite Eulerverfahren für $h = 0.1$ und $I = [0, 30]$ folgendes Ergebnis. Es ist zu erkennen, dass je weiter wir in der Zeit voran-

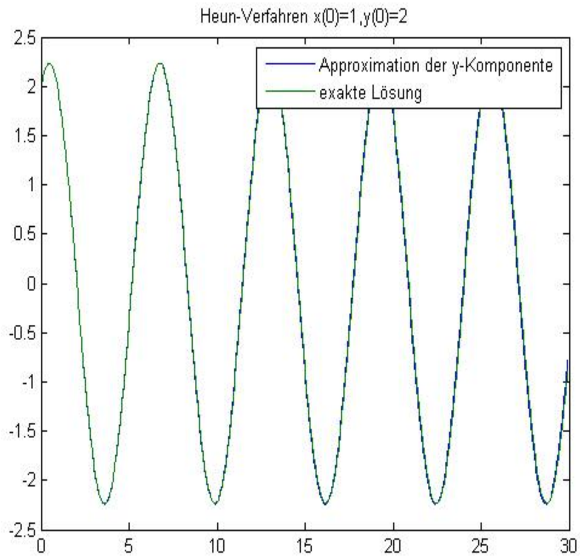




schreiten unsere Euler-Lösungen immer schlechter werden und die numerische Lösung ausreißen.

Aufgabe 1.4 (Predictor-Corrector-Verfahren
Man kann erkennen, dass die Lösung nun für beide Komponenten gegen die exakte Lösung konvergiert. Dies liegt daran, dass das Heun-Verfahren eine Kopplung aus explizitem und impliziten





Verfahren ist.

Aufgabe 1.5 (A-Stabilität)

b) Bei den meisten numerischen Verfahren für gewöhnliche DGLn können wir für festes h die n-te Iterierte in Abhängigkeit einer Funktion $g(h, \lambda)$ und der $n-1$-ten Iterierten angeben (z.B. explizit Euler angewendet auf die Dahlquistische Testgleichung $y' = \lambda y$, $y(0) = 1 \Rightarrow y_{n+1} = (1 + \lambda h)y_n$).

Die exakte Lösung der Test-DGl ist $y(t) = e^{\lambda t}$, der Realteil von λ (in unserem Fall $Re(\lambda) = \lambda$, da wir im Reellen Raum sind) entscheidet ob die exakte Lösung explodiert. Wir wollen nun, dass die numerische Lösung beschränkt bleibt, wenn es die exakte Lösung tut, daher fordert man, dass $|g(h, \lambda)| \leq 1$ für alle $\lambda \leq 0$, bzw. in unserem Spezialfall für $y' = -ky$, $y(0) = 1$, dass die numerische Lösung für alle k>0 nicht explodiert.
Der A-Stabilitätstest sagt nur aus, ob die Lösung beschränkt ist, nicht ob sie konvergiert. Betrachten wir z.B. das Verfahren $y_{n+1} = y_n$, so ist $g(h, \lambda) = 1, \forall h > 0,\ \lambda > 0$ und der Integrator besteht den A-Stabilitätstest, aber der Integrator löst nur DGLn der Form $y' = 0, y(x_0) = \tilde{x}$.