

Computerphysik

Hausarbeit 4

Friedrich Hübner 2897111

Fiona Paulus 2909625

9. Juli 2017

Der Differentialgleichungssolver

Für den DGL-Solver wird das Runge-Kutta-Verfahren mit den Fehlbergkoeffizienten und automatischer Schrittweitensteuerung aus der Vorlesung verwendet. Die Berechnung aller Werte erfolgt genau nach diesem Verfahren.

Programm (abgabe4_runge_kutta.cpp)

In dieser Datei befindet sich der DGL-Solver, der von allen anderen Programmen eingebunden wird.

Mit der Funktion 'runge_kutta_init(f, t_0, y_0)' wird der DGL-Solver initialisiert. Dabei ist $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ die Differentialgleichung ($y' = f(t, y)$), $t_0 \in \mathbb{R}$ der Anfangszeitpunkt und $y_0 \in \mathbb{R}^n$ der Anfangswert.

Mit der Funktion 'runge_kutta_iterate()' wird ein Schritt ausgeführt und die neue optimale Schrittweite berechnet.

Außerdem werden in der Datei Operatoren für Vektorarithmetik ($+, +=, -, *$) und für Ausgabe von Vektoren ($< \prec$) überladen. Zusätzlich gibt es noch Funktionen für den Betrag von Vektoren und das Skalar- und Kreuzprodukt.

Aufgabe 1 : Doppelpendel

allgemeine Hinweise

Das Programm wurde unter Linux-Mint mit "g++ -std=c++11 -g -Wall -Wextra -pedantic-errors 1.cpp -o 1.exe" kompiliert.

Aufgabenteil (a)

Aufstellen der Koordinaten:

$$\begin{aligned}x_1 &= L \sin(\phi_1) \\ y_1 &= 2L - L \cos(\phi_1) \\ x_2 &= L \sin(\phi_1) + L \sin(\phi_2) \\ y_2 &= 2L - L \cos(\phi_1) - L \cos(\phi_2)\end{aligned}$$

Daraus ergibt sich die Lagrangefunktion:

$$L(\phi_1, \phi_2, \dot{\phi}_1, \dot{\phi}_2) = 1/2 m L^2 (\dot{\phi}_1^2 + \dot{\phi}_2^2 + 2 \dot{\phi}_1 \dot{\phi}_2 \cos(\phi_1 - \phi_2)) - mgL(4 - 2 \cos(\phi_1) - \cos(\phi_2))$$

Aus dieser werden die verallgemeinerten Impulse berechnet.

$$\begin{aligned}p_1 &= \frac{\partial L}{\partial \dot{\phi}_1} = mL^2(\dot{\phi}_1 + \dot{\phi}_2 \cos(\phi_1 - \phi_2)) \\ p_2 &= \frac{\partial L}{\partial \dot{\phi}_2} = mL^2(\dot{\phi}_2 + \dot{\phi}_1 \cos(\phi_1 - \phi_2))\end{aligned}$$

Die Hamiltonfunktion setzt sich zusammen aus $\phi_1, \phi_2, \dot{\phi}_1$ und $\dot{\phi}_2$

$$H = \sum_{i=1}^2 p_i \dot{\phi}_i - L = mL^2(2\dot{\phi}_1 + \dot{\phi}_2 \cos(\phi_1 - \phi_2)\dot{\phi}_1 + (\dot{\phi}_2 + \dot{\phi}_1 \cos(\phi_1 - \phi_2))\dot{\phi}_2)$$

Durch einsetzen der Bewegungsgleichungen für ϕ_1 und ϕ_2

$$\begin{aligned}\dot{\phi}_1 &= \frac{1}{mL^2} \frac{p_1 - p_2 \cos(\phi_1 - \phi_2)}{1 + \sin(\phi_1 - \phi_2)^2} \\ \dot{\phi}_2 &= \frac{1}{mL^2} \frac{2p_2 - p_1 \cos(\phi_1 - \phi_2)}{1 + \sin(\phi_1 - \phi_2)^2}\end{aligned}$$

ergibt sich:

$$H = \frac{1}{mL^2} \frac{p_1^2 + p_2^2 - 2p_1p_2\cos(\phi_1 - \phi_2)}{1 + \sin(\phi_1 - \phi_2)^2} + mgL(4 - 2\cos(\phi_1) - \cos(\phi_2))$$

Und durch partielle Ableitung von H nach ϕ_i ergeben sich die Bewegungsgleichungen für p_1 und p_2 :

$$\dot{p}_1 = -\frac{1}{mL^2} \frac{p_1p_2\sin(\phi_1 - \phi_2)}{1 + \sin(\phi_1 - \phi_2)^2} + \frac{\sin(\phi_1 - \phi_2)\cos(\phi_1 - \phi_2)}{mL^2} \frac{p_1^2 + 2p_2^2 - 2p_1p_2\cos(\phi_1 - \phi_2)}{(1 + \sin(\phi_1 - \phi_2)^2)^2} - 2mgL\sin(\phi_1)$$

$$\dot{p}_2 = \frac{1}{mL^2} \frac{p_1p_2\sin(\phi_1 - \phi_2)}{1 + \sin(\phi_1 - \phi_2)^2} - \frac{\sin(\phi_1 - \phi_2)\cos(\phi_1 - \phi_2)}{mL^2} \frac{p_1^2 + 2p_2^2 - 2p_1p_2\cos(\phi_1 - \phi_2)}{(1 + \sin(\phi_1 - \phi_2)^2)^2} - mgL\sin(\phi_2)$$

Aufgabenteil (b)

Das durch die Hamiltonfunktion aufgestellte System von Bewegungsgleichungen werden numerisch mit dem Runge-Kutta-Verfahren numerisch gelöst.

Aufgabenteil (c)

Das in Aufgabenteil (b) geschriebene Programm wird für die Anfangsbedingungen $\phi_1 = 45^\circ$ und $\phi_2 = 135^\circ$ sowie $\phi_1 = 10^\circ$ und $\phi_2 = 5^\circ$ ausgeführt.

Daraus ergeben sich folgende Bahnkurven:

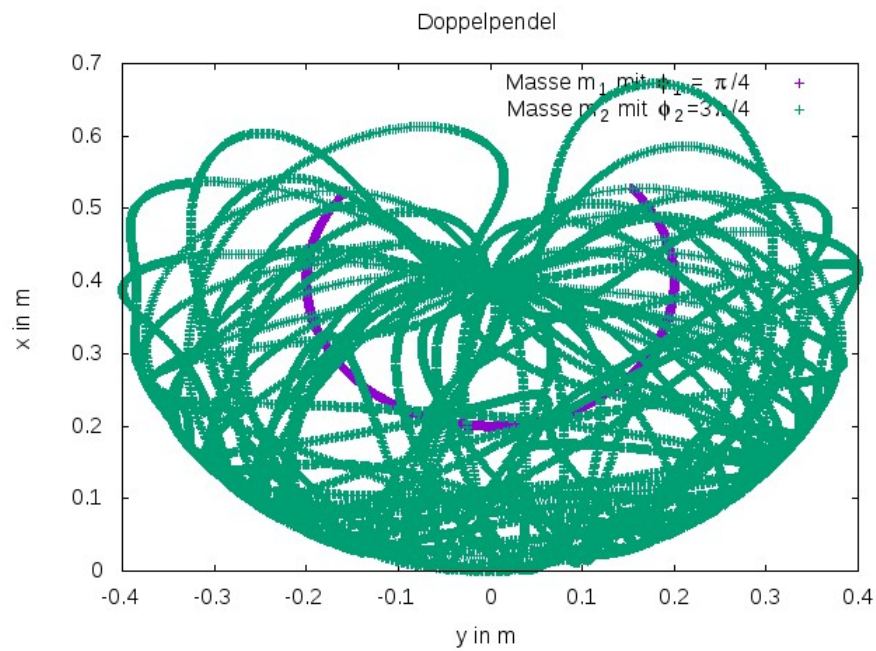


Abbildung 0.1: Bahnkurven der Massen im Doppelpendel für Anfangsbedingungen $\phi_1 = 45^\circ$ und $\phi_2 = 135^\circ$

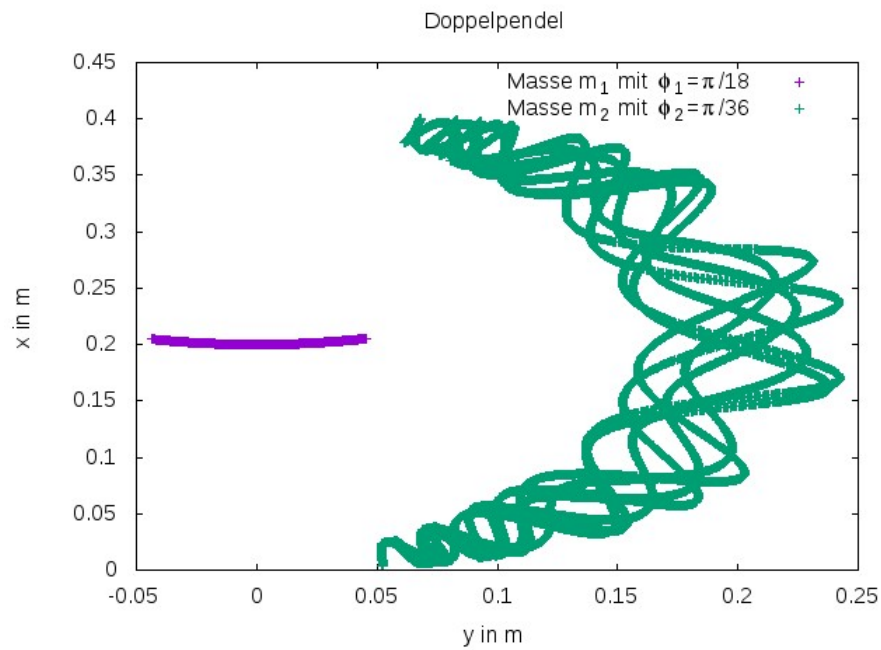


Abbildung 0.2: Bahnkurven der Massen im Doppelpendel für Anfangsbedingungen $\phi_1 = 10^\circ$ und $\phi_2 = 5^\circ$

Aufgabenteil (d)

Zur Bestimmung der zeitlichen Entwicklung der relativen Fehler der Hamiltonfunktion wird die Differenz der numerisch gelösten Hamiltonfunktion und des mithilfe der Anfangsbedingungen errechnete Wert in Abhängigkeit von der Zeit geplottet.

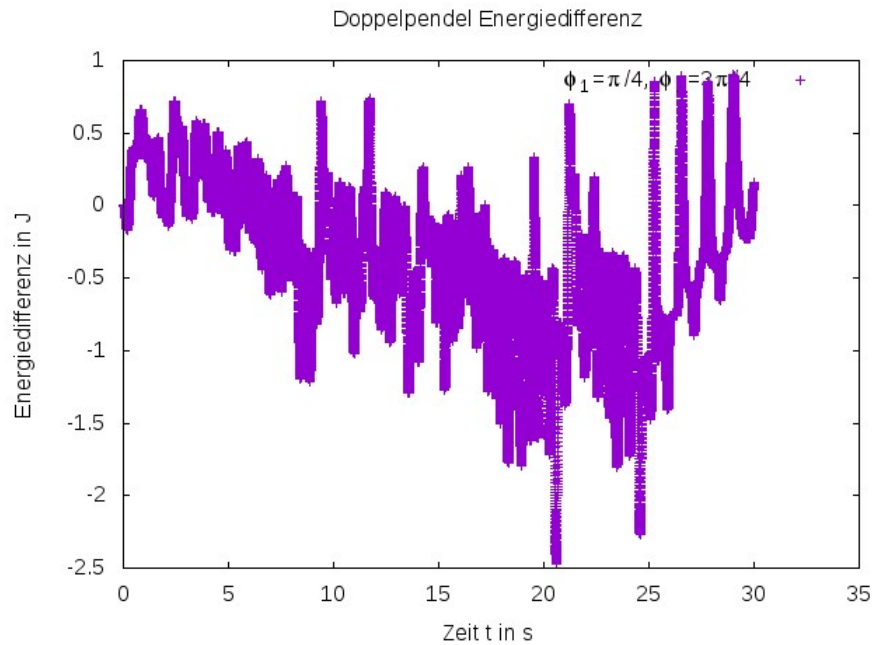


Abbildung 0.3: zeitliche Entwicklung der Abweichung der Hamiltonfunktion vom wahren Wert $H = 1.29J$ für Anfangsbedingungen $\phi_1 = 45^\circ$ und $\phi_2 = 135^\circ$

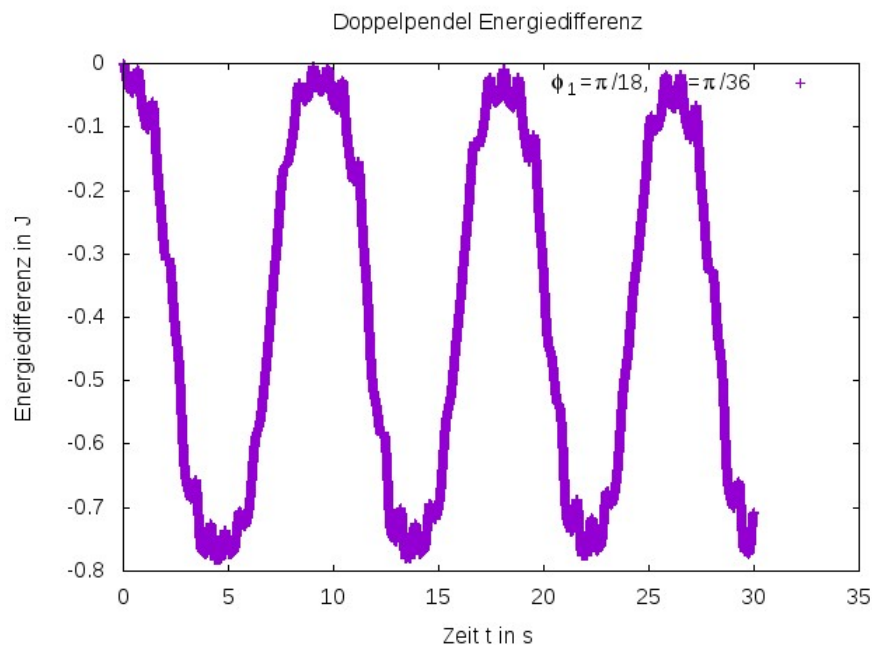


Abbildung 0.4: zeitliche Entwicklung der Abweichung der Hamiltonfunktion vom wahren Wert $H = 0.41J$ für Anfangsbedingungen $\phi_1 = 10^\circ$ und $\phi_2 = 5^\circ$

Aus den Graphen kann man erkennen, dass die Differenz des Anfangswerts der Hamiltonfunktion und der durch Runge-Kutta berechneten Werte in der gleichen Größenordnung liegen und die Abweichung beschränkt ist.

sonstige abgegebene Dateien

1.plt

Die Plot-Datei für die abgegebenen Plots.

1a.txt

Ausgabedatei des Programms "1.cpp" mit Anfangsbedingungen $\phi_1 = 45^\circ, \phi_2 = 135^\circ, p_1 = 0$ und $p_2 = 0$.

1b.txt

Ausgabedatei des Programms "1.cpp" mit Anfangsbedingungen $\phi_1 = 10^\circ, \phi_2 = 5^\circ, p_1 = 0$ und $p_2 = 0$.

Aufgabe 2: Meteorit in der Atmosphäre

allgemeine Hinweise

Das Programm wurde unter Linux-Mint mit "g++ -std=c++11 -g -Wall -Wextra -pedantic-errors 2.cpp -o 2.exe" kompiliert.

Aufgabenteil (a)

Die Bewegungsgleichung für \ddot{h} ergibt sich aus den Kräften.

$$\begin{aligned}m_k \ddot{h} &= -F_G + F \\m_k \ddot{h} &= -G \frac{m_k m_E}{(h + R_E)^2} + \frac{1}{2} c_W \rho_0 e^{-\frac{M g_0}{RT} h} A \dot{h}^2 \\ \ddot{h} &= -G \frac{m_E}{(h + R_E)^2} + \frac{1}{2 m_k} c_W \rho_0 e^{-\frac{M g_0}{RT} h} A \dot{h}^2\end{aligned}$$

mit Gravitationskonstante $G = 6.67408 \cdot 10^{-11} \frac{m^3}{kg s^2}$, Erdmasse $m_E = 5.9722 \cdot 10^{24} kg$, Erdradius $R_E = 6370 \cdot 10^3 m$, $c_W = 0.45$, Dichte am Erdboden $\rho_0 = 1.29 \frac{kg}{m^3}$, mittlerer Molarer Masse $M = 0.02896 \frac{kg}{mol}$, Erdbeschleunigung $9.81 \frac{m}{s^2}$, Gaskonstante $R = 8.314 \frac{kg m^2}{s^2 mol K}$ und der Temperatur $T = 273 K$. Die Querschnittsfläche A und die Masse m_k des Kometen berechnen sich durch:

$$\begin{aligned}A &= \frac{\pi}{2} * d^2 = \frac{\pi}{2} (1 \cdot 10^{-3})^2 = 1.57 \cdot 10^{-6} \\m_k &= \rho_k * V_k = \rho_k * \frac{\pi}{6} d^3 = 7800 \frac{kg}{m^3} \frac{\pi}{6} (1 \cdot 10^{-3})^3 = 4.084 \cdot 10^{-6}\end{aligned}$$

mit Dichte des Kometen $\rho_k = 7800 \frac{kg}{m^3}$ und Durchmesser des Kometen $d = 1 \cdot 10^{-3} m$. Durch Umwandlung der Differentialgleichung 2. Ordnung in ein Gleichungssystem 1. Ordnung ergibt sich:

$$\begin{aligned}\dot{h} &= v \\ \dot{v} &= -G \frac{m_E}{(h + R_E)^2} + \frac{1}{2 m_k} c_W \rho_0 e^{-\frac{M g_0}{RT} h} A v^2\end{aligned}$$

Aufgabenteil (b)

Die oben aufgestellten Differentialgleichungen werden mithilfe des Runge-Kutta-Verfahrens für verschiedene Anfangsbedingungen numerisch gelöst. Durch plotten der sich daraus ergebenden Geschwindigkeiten in Abhängigkeit der Höhe ergibt sich folgende Grafik:

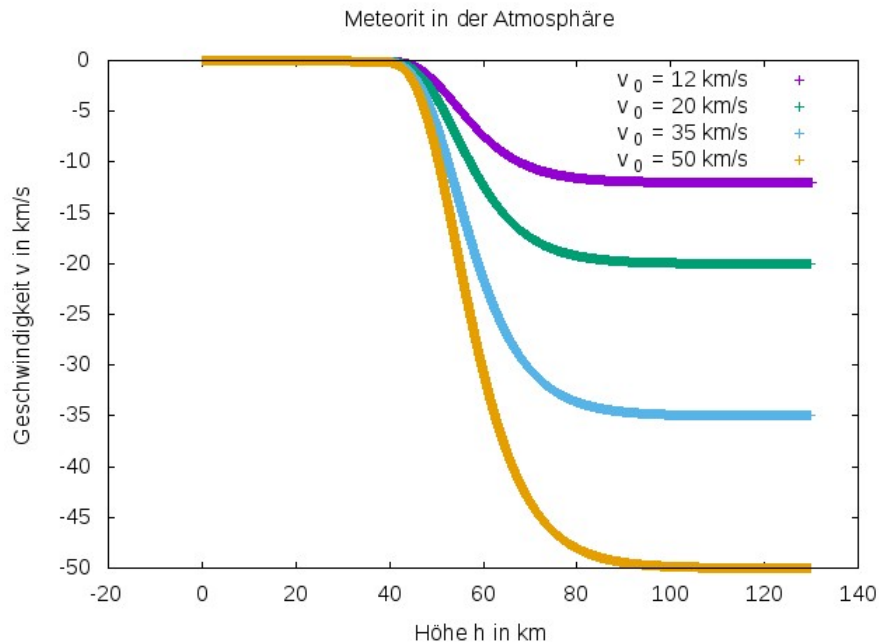


Abbildung 0.5: Geschwindigkeiten des Meteorits in Abhängigkeit der Höhe für verschiedene Anfangsgeschwindigkeiten v_0

sonstige abgegebene Dateien

2.plt

Die Plot-Datei für die abgegebenen Plots.

2a.txt

Ausgabedatei des Programms "2.cpp" mit Anfangsbedingungen $h_0 = 130000m$ und $v_0 = 12000 \frac{m}{s}$

2b.txt

Ausgabedatei des Programms "2.cpp" mit Anfangsbedingungen $h_0 = 130000m$ und $v_0 = 20000 \frac{m}{s}$

2c.txt

Ausgabedatei des Programms "2.cpp" mit Anfangsbedingungen $h_0 = 130000m$ und $v_0 = 35000\frac{m}{s}$

2d.txt

Ausgabedatei des Programms "2.cpp" mit Anfangsbedingungen $h_0 = 130000m$ und $v_0 = 50000\frac{m}{s}$

Aufgabe 3

Allgemeine Hinweise

Das Programm wurde unter Windows 10 mit "g++ -o abgabe4_3.exe -Wall -Wextra -std=c++0x -O2 -static abgabe4_3.cpp" kompiliert.

a)

Die Bewegungsgleichung lautet einfach:

$$m\ddot{\vec{x}} = e\dot{\vec{x}} \times \vec{B} \quad (0.1)$$

$$= e\dot{\vec{x}} \times \left(\frac{\mu_0}{4\pi} \frac{3\vec{x}(\vec{x} \cdot \vec{m}) - \vec{m} \cdot \vec{r}^2}{|\vec{r}|^5} \right) \quad (0.2)$$

Der Dipol wurde Richtung z-Achse gelegt: $\vec{m} = (0, 0, m)^T$.

Für $\vec{y} = (\vec{x}, \dot{\vec{x}})$ gilt nun:

$$\dot{\vec{y}} = \left(\dot{\vec{x}}, \frac{e}{m} \dot{\vec{x}} \times \left(\frac{\mu_0}{4\pi} \frac{3\vec{x}(\vec{x} \cdot \vec{m}) - \vec{m} \cdot \vec{r}^2}{|\vec{r}|^5} \right) \right) \quad (0.3)$$

b) Programm (abgabe4_3.cpp)

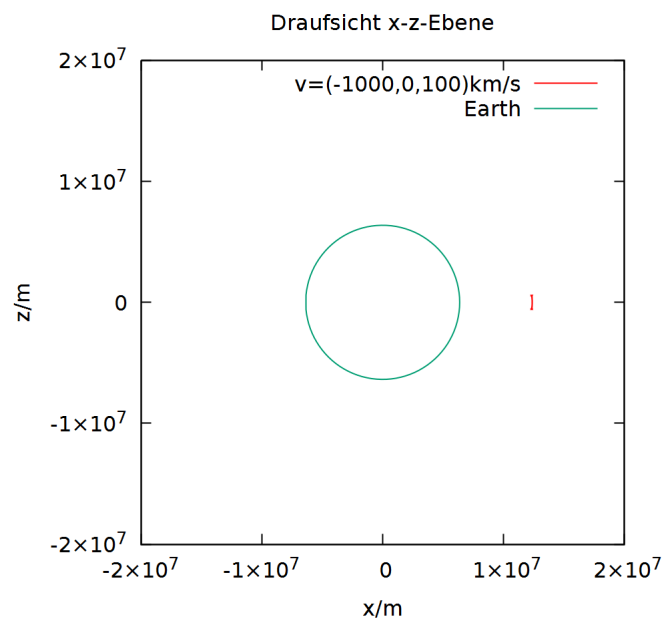
Das Programm benutzt den Fehlberg-DGL-Solver. Die verwendete Funktion f ist dabei in a) gegeben. Die Anfangsbedingungen muss man in das Programm eintragen. Das Programm simuliert 100s und gibt alle 0.1s Simulationszeit die aktuelle Zeit, Position und Geschwindigkeit aus.

c)

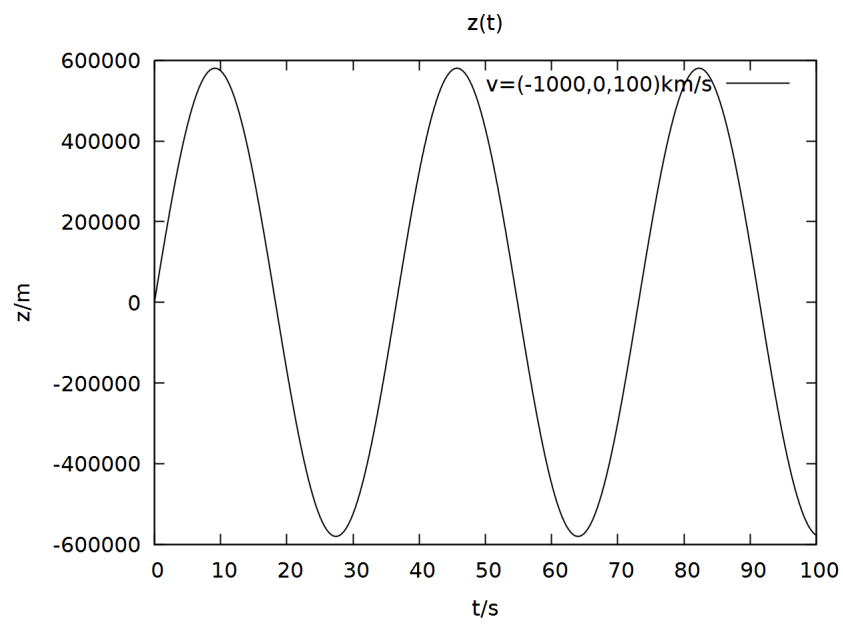
Da der Dipol in z-Richtung angeordnet ist, sind die Feldlinien am Äquator parallel zur z-Achse. Also startet das Proton mit $\vec{v} = (-1000, 0, v_z)^T$.

1. $v_z = 100\text{km/s}$

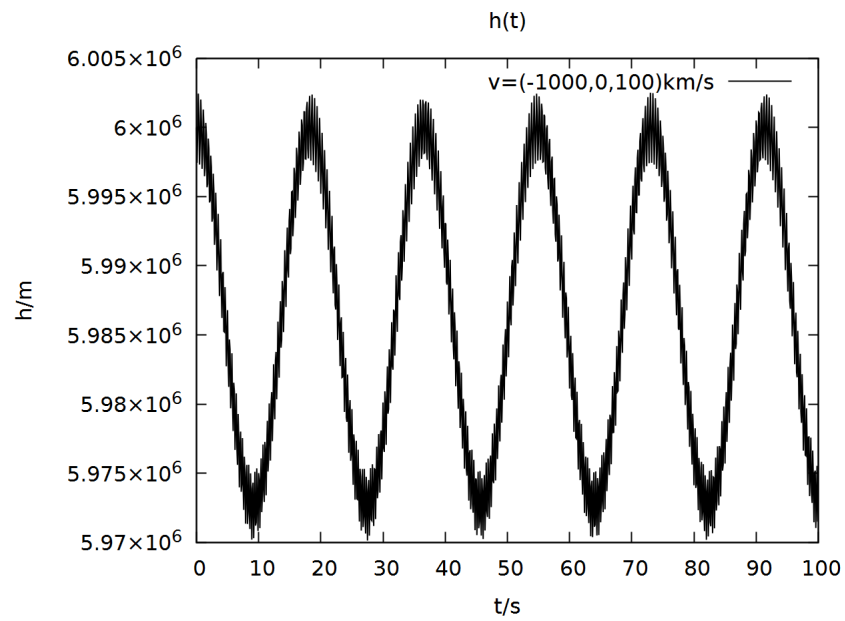
i)



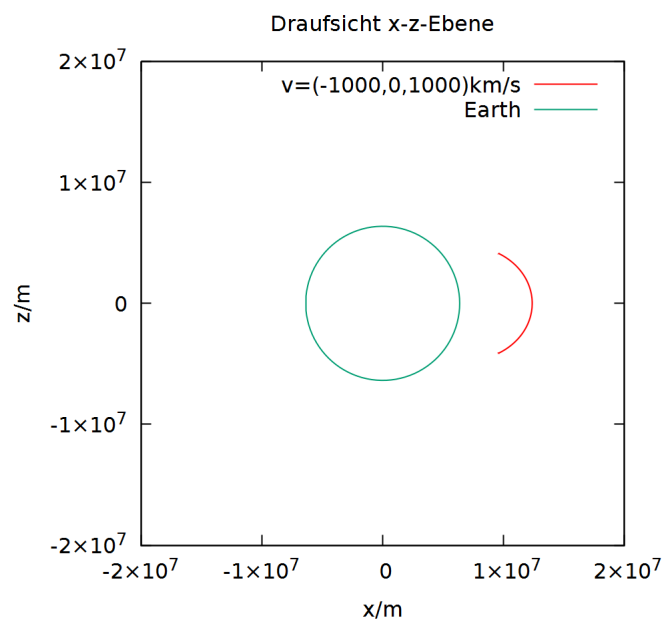
ii)



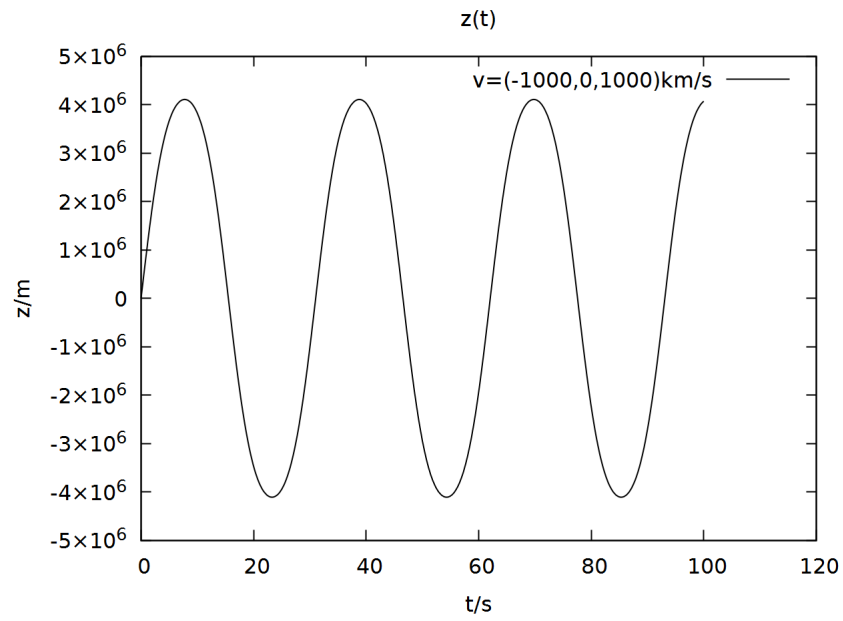
iii)

2. $v_z = 1000 \text{ km/s}$

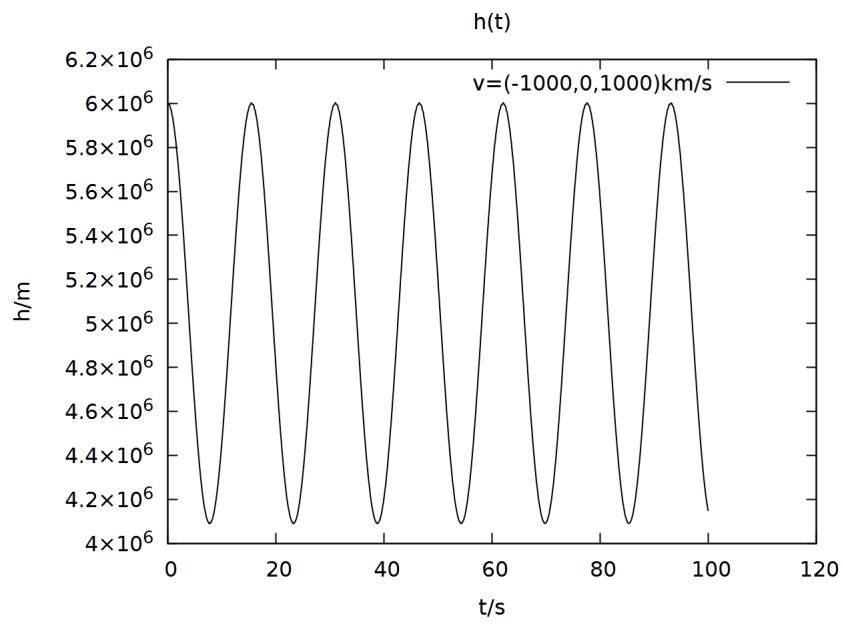
i)



ii)

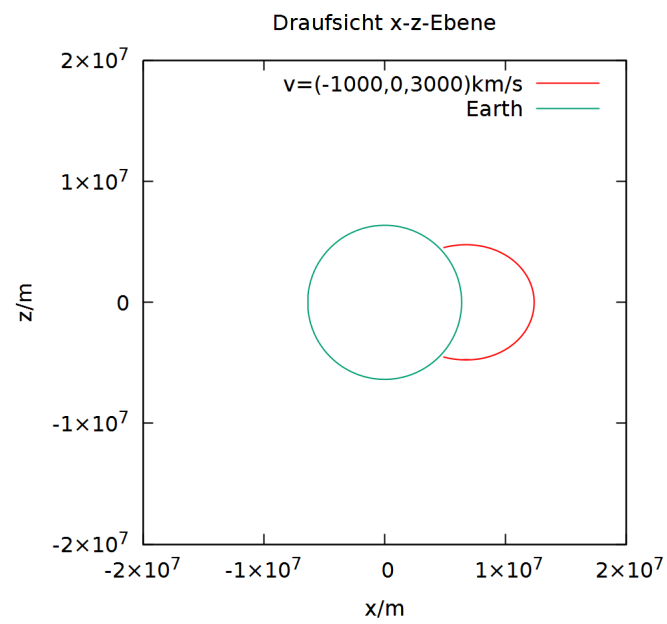


iii)

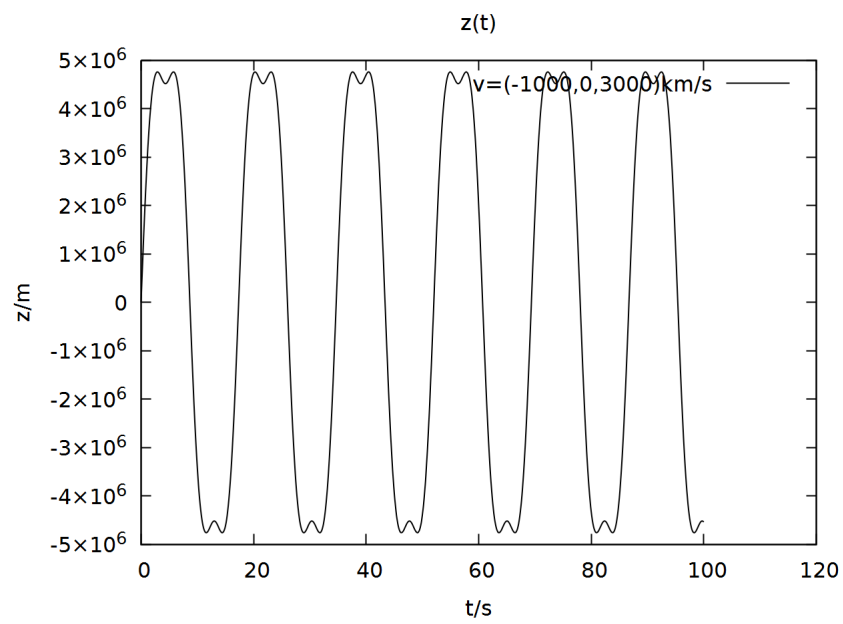


3. $v_z = 3000 \text{ km/s}$

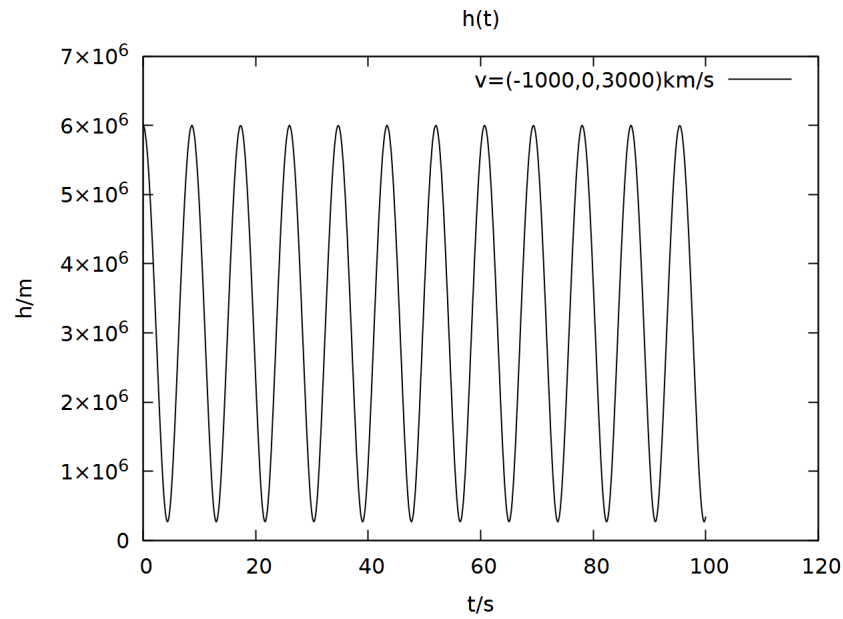
i)



ii)



iii)



Sonstige abgegebene Dateien

`plot_i.plt`, `plot_ii.plt`, `plot_iii.plt`

Die Plot-Dateien für die Diagramme für jeweils Fall i), ii), iii)

`1.txt`, `2.txt`, `3.txt`

Die Ausgabedateien für die einzelnen Anfangswerte