

## Trifft uns der Komet?

### *Teilnehmer:*

|                        |                          |
|------------------------|--------------------------|
| Paul Barth             | Herder-Gymnasium         |
| Hristo Boyadzhiev      | Käthe-Kollwitz-Gymnasium |
| Batkhaan Erdenetsetseg | Heinrich-Hertz-Gymnasium |
| Roxana Esmaili         | Herder-Gymnasium         |
| Bryan Oppong-Boateng   | Andreas-Gymnasium        |
| Felix Roth             | Herder-Gymnasium         |
| Julius Paul Rußmann    | Herder-Gymnasium         |
| Jean-Pierre Walter     | Herder-Gymnasium         |
| Bela Tim Wiethüchter   | Andreas-Gymnasium        |

### *Gruppenleiter:*

René Lamour, Humboldt-Universität zu Berlin  
Nina Fleischer, Humboldt-Universität zu Berlin

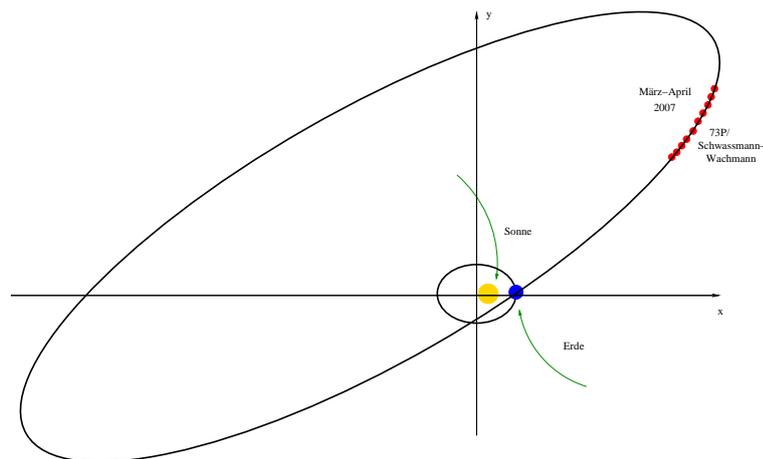


## Einleitung

„200 Meter hohe Flutwelle im Atlantik? Komet rast auf Erde zu - Forscher in großer Sorge“, titelte Bild.de am 01.04.2007 und bezog sich dabei auf den Kometen „73 P Schwassmann-Wachmann“. Teile dieses Kometen könnten, so Bild.de weiter, „wie eine ungeheure Bombe wirken“.

Somit stellt sich die Frage: Kann der Komet uns tatsächlich treffen? Haben die Umlaufbahnen der Erde und des Kometen also tatsächlich Schnittpunkte?

Dieser Problematik kann man sich durch die Modellierung eben jener Umlaufbahnen annähern. Nach dem ersten Keplerschen Gesetz befinden sich Planeten, Asteroide und Kometen auf Kegelschnittbahnen; wiederkehrende auf Ellipsen. Das Zentralgestirn - im Fall der Erde die Sonne - befindet sich in einem Brennpunkt der Ellipsen.



Umlaufbahnen von Komet und Erde

Die Umlaufbahnen lassen sich anhand von Positionsdaten des Kometen vereinfacht mit den folgenden Gleichungen beschreiben:

$$\begin{array}{l} \text{Erde-Sonne} \\ \left(\frac{x}{1}\right)^2 + \left(\frac{y}{0.999}\right)^2 = 1 \\ \text{Komet} \\ \left(\frac{x+y-2}{6}\right)^2 + \left(\frac{x-y+2}{3}\right)^2 = 1 \end{array}$$

Bei diesen Gleichungen handelt es sich um ein mathematisches Modell. Es beschreibt also einen realen Prozess und berücksichtigt nur die wesentlichen Komponenten und Eigenschaften, um das Problem mathematisch so weit wie möglich zu vereinfachen. In diesem expliziten Fall werden lediglich zweidimensionale Projektionen der Bahnen betrachtet. Des Weiteren werden der Einfluss der Gravitation der Erde auf den Kometen und die exakte Position der einzelnen Himmelskörper auf ihren jeweiligen Umlaufbahnen vernachlässigt.

Es wird also nicht untersucht, ob Erde und Komet tatsächlich kollidieren, sondern lediglich ob sich die jeweiligen Umlaufbahnen im zweidimensionalen Raum kreuzen.

Die oben genannten Gleichungen können durch ein nichtlineares Gleichungssystem der folgenden Form

$$f(x, y) := \begin{pmatrix} f_1(x, y) \\ f_2(x, y) \end{pmatrix} = 0$$

mit

$$\begin{array}{l} f_1(x, y) := \left(\frac{x}{1}\right)^2 + \left(\frac{y}{0.999}\right)^2 - 1 \\ f_2(x, y) := \left(\frac{x+y-2}{6}\right)^2 + \left(\frac{x-y+2}{3}\right)^2 - 1 \end{array}$$

beschrieben werden.

Allgemein sollen im Folgenden Gleichungen der Form

$$f(x) = \begin{pmatrix} f_1(x_1, x_2, \dots, x_n) \\ f_2(x_1, x_2, \dots, x_n) \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) \end{pmatrix} = 0$$

untersucht und gelöst werden.

Hierzu soll nun zur Problemlösung mehrere Verfahren, wie insbesondere das „Newton-Verfahren“, herangezogen werden.

### Nullstellenberechnung durch Iteration im eindimensionalen Raum ( $n = 1$ )

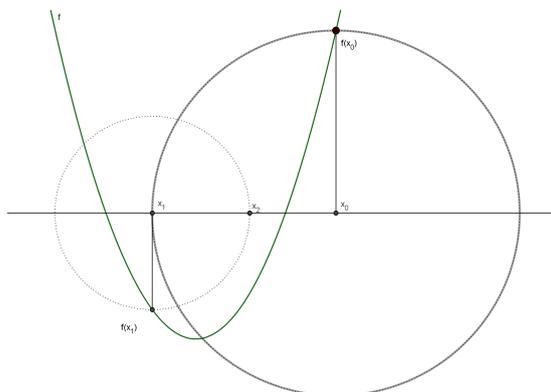
Unser Ziel ist es, für eine gegebene Funktion  $f : A \subset \mathbb{R} \rightarrow \mathbb{R}$  die Nullstellen zu ermitteln. Für Polynome 1. - 4. Grades existieren noch allgemeine Lösungsformeln, für Polynome höheren Grades gibt es keine allgemeinen Lösungsformeln mehr. Wir suchen also einen Weg, auch für Gleichungen, die keine explizite Lösungsformeln haben, eine Lösung näherungsweise zu bestimmen. Dazu werden so genannte Iterationsverfahren verwendet. Das sind Gleichungen der Form  $x_{i+1} = \Phi(x_i)$ .

$\Phi : A \rightarrow A$  ist dabei eine Kontraktion auf  $A$ , das bedeutet

1.  $\Phi(A) \subseteq A$
2.  $\forall x_1, x_2 \in A \quad |\Phi(x_1) - \Phi(x_2)| \leq K|x_1 - x_2|$  mit Lipschitz-Konstante  $0 \leq K < 1$  ( $K$  wie Lipschitz :)

Es sei die Nullstelle  $x_*$  der Funktion  $f$  in dem Intervall  $[a, b]$  bzw. in einer Umgebung des Startwerts  $x_0$  zu bestimmen.

- **einfache Iteration:** Die Nullstelle wird durch  $x_{i+1} := x_i - f(x_i)$  angenähert.

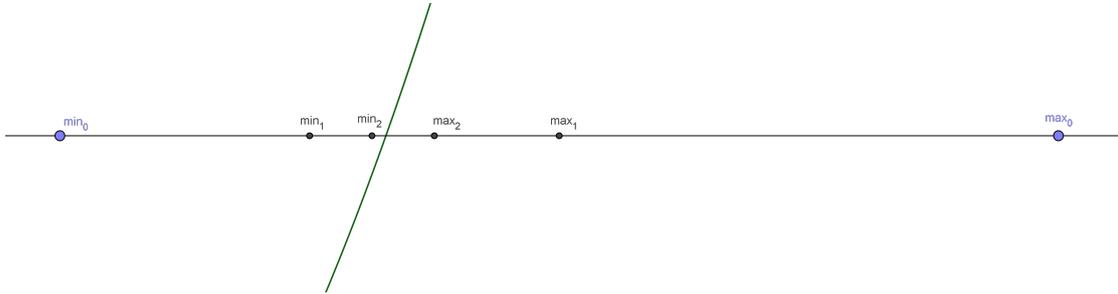


Nach Durchführung des Verfahrens mit Hilfe von Python erhält man für den Startwert  $x_0 = 0$  mit der Beispielfunktion  $f$  mit  $f(x) = x^2 - 2$  folgende Testwerte:

| Iterationsschritt | iterierter Wert    |
|-------------------|--------------------|
| 1                 | 1.0                |
| 2                 | 1.5                |
| 3                 | 1.375              |
| 4                 | 1.4296875          |
| 5                 | 1.407684326171875  |
| 6                 | 1.4168967450968921 |
| 7                 | 1.4130985519638084 |
| 8                 | 1.4146747931827024 |

• **Bisektionsverfahren:**

**Satz 1** (Zwischenwertsatz). Sei  $f : [a, b] \rightarrow \mathbb{R}$  punktweise stetig auf  $[a, b]$  mit  $f(a) < 0$ ,  $f(b) > 0$  (oder umgekehrt), dann existiert ein  $x_0 \in [a, b]$  mit  $f(x_0) = 0$ .



**Beweis:** Definiere eine Folge  $\{[a_i, b_i]\}$  für  $i = 1, 2, \dots, n$  mit  $f(a_i) \leq 0$ ,  $f(b_i) \geq 0$  und  $b_i - a_i = \frac{a-b}{2^i}$ . Es sei  $a_0 = a$  und  $b_0 = b$  und  $x_i = \frac{a_i + b_i}{2}$ .

1. Für  $f(x_i) \leq 0$  wird  $a_{i+1} = x_i$  und  $b_{i+1} = b_i$ ,
2. Für  $f(x_i) > 0$  wird  $a_{i+1} = a_i$  und  $b_{i+1} = x_i$ .

Somit findet man die Nullstelle, da  $a_i$  monoton wachsend und nach oben beschränkt ( $g = \tilde{a}$ ) und  $b_n$  monoton fallend und nach unten beschränkt ( $g = \tilde{b}$ ) ist. Da  $\tilde{a} = \tilde{b} =: x_0$  und  $f$  stetig,

$$\left. \begin{aligned} f(x_0) &= \lim_{i \rightarrow \infty} f(a_i) \leq 0 \\ f(x_0) &= \lim_{i \rightarrow \infty} f(b_i) \geq 0 \end{aligned} \right\} \Rightarrow f(x_0) = 0.$$

□

Nach Durchführung des Verfahrens mit Hilfe von Python erhält man für die Intervallgrenzen  $a = 0$  und  $b = 6$  mit der Beispielfunktion  $f(x) = x^2 - 2$  folgende Testwerte:

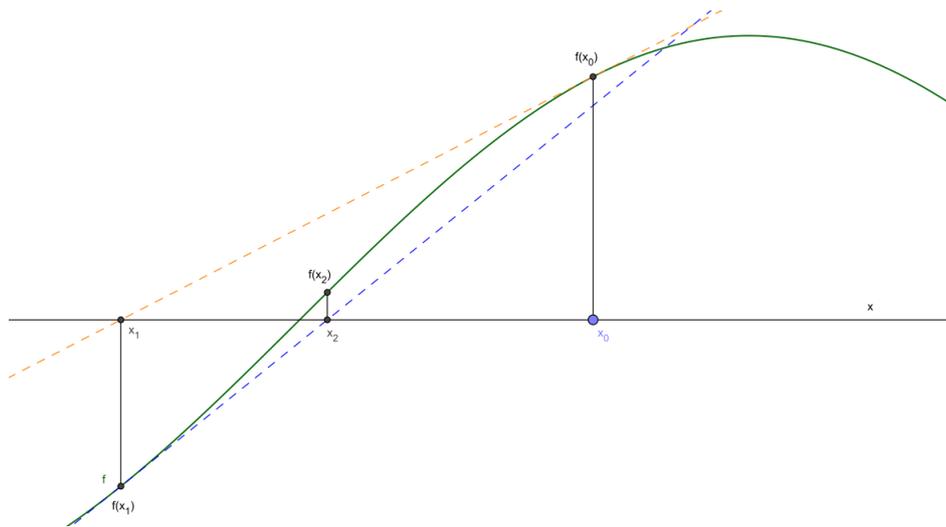
| Iterationsschritt | iterierter Wert  |
|-------------------|------------------|
| 1                 | 3.0              |
| 2                 | 1.5              |
| 3                 | 0.75             |
| 4                 | 1.125            |
| 5                 | 1.3125           |
| 6                 | 1.40625          |
| 7                 | 1.453125         |
| 8                 | 1.4296875        |
| 9                 | 1.41796875       |
| 10                | 1.412109375      |
| 11                | 1.4150390625     |
| 12                | 1.41357421875    |
| 13                | 1.414306640625   |
| 14                | 1.4139404296875  |
| 15                | 1.41412353515625 |

• **Newton-Verfahren**

**Satz 2** (Newton-Iteration). Sei  $A$  abgeschlossen und  $f$  auf  $A$  stetig und differenzierbar. Sei  $x_*$  eine Nullstelle von  $f$  auf  $A$  und  $f'(x_*) \neq 0$ . Dann existiert  $I = [x_* - s; x_* + s]$ , so dass die Newton-Iteration für jedes  $x_0 \in I$  verhältnismäßig sehr schnell gegen  $x_*$  konvergiert.

Die Nullstelle der Tangente an dem Punkt  $(x_i, f(x_i))$  ist der Wert der nächsten Iteration:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}.$$



**Überlegung.** Um die Vorschrift besser zu verstehen, sollte man sich einen Iterationsschritt vor Augen führen. Es gilt nach dem Steigungsdreieck:  $f'(x_0) = \frac{f(x_0)}{x_0 - x_1} \iff \frac{f(x_0)}{f'(x_0)} = x_0 - x_1 \iff x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$ . Dies lässt sich nun für jede beliebige  $i$ -te Iteration übertragen.

Nach Durchführung des Verfahrens mit Hilfe von Python erhält man mit der Beispielfunktion  $f(x) = x^2 - 2$  und dem Startwert  $x_\star = 0.1$  folgende Testwerte:

| Iterationsschritt | iterierter Wert    |
|-------------------|--------------------|
| 1                 | 10.044530508184575 |
| 2                 | 5.124284495408854  |
| 3                 | 2.7584766561773937 |
| 4                 | 1.7422672026786925 |
| 5                 | 1.4452477367580894 |
| 6                 | 1.4145622136776876 |
| 7                 | 1.4142137807876884 |
| 8                 | 1.414213562483036  |
| 9                 | 1.4142135623731504 |
| 10                | 1.4142135623731504 |

## Mehrdimensionale Grundlagen

Um diese eindimensionalen Iterationsverfahren ins Mehrdimensionale zu bringen, werden wir nicht mehr Gleichungen, sondern Gleichungssysteme lösen. Um uns die Arbeit mit diesen zu strukturieren und erleichtern, führen wir einige Definitionen ein.

### Definition 1. (Vektor)

Ein  $n$ -Tupel reeller Zahlen

$$v := \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \quad v_1, v_2, \dots, v_n \in \mathbb{R} \quad (1)$$

heißt reeller Vektor mit  $n$  Einträgen.  $\mathbb{R}^n$  bezeichnet die Menge aller Vektoren mit  $n$  Einträgen.

**Definition 2.** (Matrix)

Für die Zahlen  $a_{ij} \in \mathbb{R}$ ,  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, n$  heißt das Zahlenschema

$$A := \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix} := \{a_{ij}\}_{\substack{i=1,2,\dots,m \\ j=1,2,\dots,n}} := \{a_{ij}\} \quad (2)$$

eine Matrix des Formats  $m \times n$ .  $\mathbb{R}^{m,n}$  bezeichnet die Menge aller Matrizen des Formats  $m \times n$ .

Im Wesentlichen ist die Matrix eine Erweiterung des Vektors, in der es mehrere Spalten geben kann. Somit kann jeder Vektor als Spezialfall einer Matrix mit einer Spalte betrachtet werden. Ferner werden für unseren Zweck der Lösung von Gleichungssystemen nur quadratische Matrizen betrachtet, wo also  $m = n$  gilt.

**Definition 3.** (Vektorraumoperationen)

Gegeben seien  $A := \{a_{ij}\}, B := \{b_{ij}\} \in \mathbb{R}^{m,n}$ ,  $\alpha \in \mathbb{R}$ .

$$\begin{aligned} A + B &:= \{a_{ij} + b_{ij}\} \\ \alpha A &:= \{\alpha a_{ij}\} \end{aligned}$$

**Definition 4.** (Matrixmultiplikation)

Gegeben sei  $A := \{a_{ij}\} \in \mathbb{R}^{m,k}$  und  $B := \{b_{jl}\} \in \mathbb{R}^{k,n}$

$$AB := \left\{ \sum_{j=1}^k a_{ij} \cdot b_{jl} \right\}_{\substack{i=1,\dots,m \\ l=1,\dots,n}} \in \mathbb{R}^{m,n} \quad (3)$$

Um auf unseren Spezialfall der quadratischen Matrizen und die Betrachtung von Vektoren als Matrizen zurückzukommen gilt also für  $A \in \mathbb{R}^{n,n}$  und  $v \in \mathbb{R}^n$ :

$$Av = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \cdot \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^n a_{1j} \cdot v_j \\ \sum_{j=1}^n a_{2j} \cdot v_j \\ \vdots \\ \sum_{j=1}^n a_{nj} \cdot v_j \end{pmatrix} \in \mathbb{R}^n \quad (4)$$

Man betrachte nun das lineare Gleichungssystem:

$$\begin{aligned} a_{11} \cdot x_1 + a_{12} \cdot x_2 + \dots + a_{1n} \cdot x_n &= b_1 \\ a_{21} \cdot x_1 + a_{22} \cdot x_2 + \dots + a_{2n} \cdot x_n &= b_2 \\ &\vdots \\ a_{n1} \cdot x_1 + a_{n2} \cdot x_2 + \dots + a_{nn} \cdot x_n &= b_n \end{aligned}$$

Fassen wir alle  $a_{i,j}$  in einer Matrix und alle  $x_i$  bzw.  $b_i$  in je einem Vektor zusammen, so sieht dasselbe Gleichungssystem folgendermaßen aus:

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \quad (5)$$

Somit können wir lineare Gleichungssysteme als Matrixgleichungen behandeln.

In einer linearen Gleichung  $ax = b$  lässt sich  $x$  ermitteln, indem man beide Seiten mit  $\frac{1}{a} = a^{-1}$  multipliziert. Im Vektorraum der reellen Zahlen ist  $a^{-1}$  das inverse Element von  $a$ , also die Zahl, für welche  $a^{-1} \cdot a = 1$  gilt. Um analog mit Matrizen arbeiten zu können, müssen wir wissen, was in ihrem Raum der  $1 \in \mathbb{R}$  entspricht und wie das inverse Element definiert ist.

**Definition 5.** (Einheitsmatrix) Die Matrix  $I_n \in \mathbb{R}^n$ ,  $n$  mit:

$$I_n := \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \quad (6)$$

heißt Einheitsmatrix vom Format  $n \times n$ .

Diese hat die Eigenschaft  $AI_n = A = I_n A$  für alle  $A \in \mathbb{R}^{m,n}$ .

Durch das Einselement lässt sich nun zu jedem (invertierbaren) Element das inverse Element definieren.

**Definition 6.** (Inverse Matrix)

Eine quadratische Matrix  $A \in \mathbb{R}^{n,n}$  heißt invertierbar, wenn eine Matrix  $B \in \mathbb{R}^{n,n}$  existiert, sodass

$$AB = I_n \quad (7)$$

$B$  heißt dann die Inverse zu  $A$  und wird mit  $A^{-1}$  bezeichnet.

Kriterien zur Ermittlung der Invertierbarkeit werden hier nicht diskutiert. Wir betrachten ferner nur invertierbare Matrizen.

Für unsere Beispielsgleichung gilt also:

$$\begin{aligned} Ax &= b \\ A^{-1}Ax &= A^{-1}b \\ I_n x &= A^{-1}b \\ x &= A^{-1}b \end{aligned}$$

Um Längen von Vektoren bzw. Abstände zwischen Vektoren zu messen, definieren wir die dem eindimensionalen Betrag entsprechende Normabbildung.

**Definition 7.** (Norm eines Vektors)

Eine Abbildung:

$$\begin{aligned} \mathbb{R}^n &\rightarrow \mathbb{R}_0^+ \\ v &\mapsto \|v\| \end{aligned}$$

heißt Norm, wenn sie folgende Eigenschaften erfüllt:

1.  $\|v\| \geq 0$ ,  $\|v\| = 0 \Leftrightarrow v = 0$
2.  $\|\alpha v\| = |\alpha| \cdot \|v\| \quad \alpha \in \mathbb{R}, v \in \mathbb{R}^n$
3.  $\|v + w\| \leq \|v\| + \|w\|$

Um mehrdimensionale Funktionen zu differenzieren, müssen wir diese partiell differenzieren.

**Definition 8.** (Partielle Ableitung)

Sei  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  eine Funktion und  $a = (a_1, \dots, a_n)$  ein Element in  $\mathbb{R}^n$ . Der Grenzwert

$$\frac{\partial f}{\partial x_i}(a) := \lim_{h \rightarrow 0} \frac{f(a_1, \dots, a_i + h, \dots, a_n) - f(a_1, \dots, a_i, \dots, a_n)}{h} \quad (8)$$

heißt partielle Ableitung von  $f$  nach der  $i$ -ten Variablen  $x_i$  im Punkt  $a$ .

### Mehrdimensionales Newtonverfahren

**Herleitung.** Sei  $g : \mathbb{R} \rightarrow \mathbb{R}$  mit  $g(x) = 0$  ein eindimensionales Gleichungssystem. Die Iterationsvorschrift  $x_{i+1} = x_i - \frac{g(x_i)}{g'(x_i)}$  für ein solches Gleichungssystem erhalten wir aus der Gleichung der Tangente  $t(x) = g(x) + g'(x) \cdot (x - x_i)$  an den Punkt  $(x_i | g(x_i))$ , die die Steigung  $g'(x_i)$  hat.

Sei  $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$  mit

$$f(x) = \begin{pmatrix} f_1(x_1, x_2, \dots, x_n) \\ f_2(x_1, x_2, \dots, x_n) \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) \end{pmatrix} = 0$$

nun ein mehrdimensionales Gleichungssystem in  $n$  Variablen  $x_1, x_2, \dots, x_n$ . Die Gleichung der Linearisierung in einen Vektor  $x_i = (x_1^i, x_2^i, \dots, x_n^i)$  lautet analog zur Tangentengleichung eines eindimensionalen Gleichungssystems

$$t(x) = \underbrace{f(x_i)}_{\in \mathbb{R}^n} + \underbrace{Df(x_i)}_{\in \mathbb{R}^{n,n}} \cdot \underbrace{(x - x_i)}_{\in \mathbb{R}^n}. \quad (9)$$

Dabei ist  $Df$  eine sogenannte Jacobi-Matrix, bei der der Eintrag  $c_{ij} = \frac{\partial f_i}{\partial x_j}$  die partielle Ableitung von  $f_i$  nach  $x_j$  ist:

$$Df = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{pmatrix} \quad (10)$$

Für die Bestimmung der Nullstelle der Linearisierung ergibt sich folgende Gleichung:

$$\begin{aligned} 0 &= f(x_0) + Df(x_0)(x - x_0) && | - f(x_0) \\ -f(x_0) &= Df(x_0)(x - x_0) && | \cdot Df(x_0)^{-1} \\ -Df(x_0)^{-1}f(x_0) &= x - x_0 && | + x_0 \\ x_0 - Df(x_0)^{-1}f(x_0) &= x \end{aligned} \quad (11)$$

Also ist

$$x_{i+1} = x_i - Df(x_i)^{-1}f(x_i).$$

Da sich die Inverse einer Matrix nicht besonders effizient berechnen lässt, empfiehlt es sich den Ansatz aus Gleichung (??) zu verwenden

$$\begin{aligned} -f(x_i) &= Df(x_i)(x_{i+1} - x_i) && | \cdot (-1) \\ f(x_i) &= Df(x_i)\Delta x_i \\ \text{mit } \Delta x_i &= x_i - x_{i+1} \iff x_{i+1} = x_i - \Delta x_i \end{aligned}$$

Dementsprechend ergeben sich die Beziehungen

$$Df(x_i)\Delta x_i = f(x_i) \quad (12)$$

und

$$x_{i+1} = x_i - \Delta x_i. \quad (13)$$

(??) ist dabei ein lineares Gleichungssystem der Form  $Ax = b$ . Das Newton-Verfahren für mehrdimensionale Gleichungssysteme besteht also darin, in jeder Iteration das lineare Gleichungssystem (??) zu lösen, und die Lösung in Gleichung (??) einzusetzen.

**Programmierung.** Zur Programmierung verwendeten wir Python. Zur Nutzung von Matrizen und Vektoren nutzen wir die freie Bibliothek `numpy`. Aus dieser Bibliothek haben wir unter anderem ein Skript zur Lösung von linearen Gleichungssystemen verwendet, so dass  $\Delta x_i$  und damit  $x_{i+1}$  berechnet werden konnte. Dies wurde dann sehr oft wiederholt, womit eine sehr hohe Genauigkeit erreicht wurde. Durch das Speichern der Vektoren in einer Liste konnten wir mithilfe des Satzes zur Konvergenzordnung

$$\|x_{i+1} - x_i\| \leq c \|x_i - x_{i-1}\|^p \quad (c < 1 \text{ für } p = 1) \quad (14)$$

und den Gleichungen

$$\frac{\|x_{i+1} - x_i\|}{\|x_i - x_{i-1}\|} = \left( \frac{\|x_i - x_{i-1}\|}{\|x_{i-1} - x_{i-2}\|} \right)^p \quad (15)$$

und

$$p = \frac{\ln\left(\frac{\|x_{i+1}-x_i\|}{\|x_i-x_{i-1}\|}\right)}{\ln\left(\frac{\|x_i-x_{i-1}\|}{\|x_{i-1}-x_{i-2}\|}\right)} \quad (16)$$

die Konvergenzordnung feststellen, sie lag bei 2.

Die Jacobi-Matrix ist für die Lösung des Kometenproblems fest einprogrammiert und lautet

$$\begin{pmatrix} 2x & \frac{2y}{0.998001} \\ \frac{x+y-2}{18} + \frac{2(x-y+2)}{9} & \frac{x+y-2}{18} - \frac{2(x-y+2)}{9} \end{pmatrix}.$$

Somit ergab sich der folgende Programmcode:

```
from numpy.ma import zeros, log
from numpy.linalg import solve, norm

def f(x):
    x1 = x[0]
    x2 = x[1]
    y = zeros(2)
    y[0]=x1**2+(x2/0.999)**2-1
    y[1]=((x1+x2-2)/6)**2+((x1-x2+2)/3)**2-1
    return y

def Df(z):
    x=z[0]
    y=z[1]
```

```

df=zeros((2,2))
df[0,0]=2*x
df[0,1]=(2/0.998001)*y
a=(1/3.)*((x+y-2.)/6)
b=(2/3.)*((x-y+2.)/3)
df[1,0]=a+b
df[1,1]=a-b
return df

def order(liste):
    c = norm(liste[len(liste)-1]-liste[len(liste)-2])/\
        norm(liste[len(liste)-2]-liste[len(liste)-3])
    d = norm(liste[len(liste)-2]-liste[len(liste)-3])/\
        norm(liste[len(liste)-3]-liste[len(liste)-4])
    return log(c)/log(d)

x=zeros(2)
x[0]=1.
x[1]=1.

i=0
l=[x]

while norm(f(x))>1e-14:
    dx = solve(Df(x),f(x))
    print 'dx: ',dx
    print 'Norm delta x: ',norm(dx)
    x=x-dx
    print 'x: ', x
    l.append(x)
    print 'Norm von x: ',norm(x)
    if i > 2:
        print 'Ordnung: ', order(l), '\n'
    i+=1

```

**Lösung.** Wenden wir nun das geschriebene Programm auf das Kometenproblem an, erhalten wir mit den Startwerten  $x_0 = 1$  und  $y_0 = 1$  sowie der gewählten Genauigkeit  $10^{-14}$  die erste Lösung ( $x \approx 0.99927129, y \approx 0.0381322$ ). Für die Startwerte  $x_0 = -1$  und  $y_0 = -1$  erhalten wir dann auch die zweite Lösung ( $x \approx -0.4121167, y \approx -0.91021994$ ) (vgl. auch Abb. ??). Insgesamt lautet also die Lösungsmenge

$$L = \{(x \approx 0.99927129, y \approx 0.0381322); (x \approx -0.4121167, y \approx -0.91021994)\}.$$

**Evaluierung.** Obwohl wir beide Schnittpunkte des Kometen und der Erde ermitteln konnten, können sich durchaus einige Probleme bei der Iteration ergeben. Einerseits lässt sich bei ungünstigen Startwerten keine Lösung ermitteln, da die Iteration dann nicht konvergiert. So ergibt sich mit den Startwerten  $x_0 = 0$  und  $y_0 = 0$  eine singuläre Matrix, so dass die Iteration abbricht.

Ferner ist die Konvergenzordnung von  $p = 2$  nicht immer, sondern erst ab einer gewissen Anzahl von Iterationsschritten gegeben, sodass die Iteration möglicherweise anfangs nur sehr langsam konvergiert.