

Aspekte der mathematischen Modellierung in der Biologie

Wiebke Johanna Sunhild Trost
Universität Kassel
Fachbereich 17, Mathematik

1. September, 2005

Inhaltsverzeichnis

0	Einleitung	2
1	Populationsprozesse	3
1.1	Wachstumsprozesse einer Art	3
1.1.1	Exponentielles Wachstum	4
1.1.2	Logistisches Wachstum	10
1.1.3	Wachstum einer Art bei regelmäßigen Dezimierungseingriffen . .	15
1.2	Populationsmodelle für zwei Arten in einem Lebensraum	17
1.2.1	Räuber-Beute-Modelle	17
1.2.2	Wachstumsmodelle konkurrierender Arten	27
1.2.3	Wachstumsmodelle für Arten in Symbiose	28
1.3	Wachstumsmodelle für mehrere Arten in Wechselbeziehungen	30
2	Neurophysiologie	32
2.1	Einführung in die Thematik	32
2.2	Das Modell: VCON - Voltage Controlled Oscillator Neuron	34
2.2.1	Ein einzelnes Neuron	35
2.2.2	Neuronen-Netzwerke	45
2.3	Andere Neuronen-Modelle	45
3	Anwendungen von Markov-Ketten in der Biologie	46
3.1	Einführung in die Theorie der Markov-Ketten	46
3.2	Simulation und Anwendungen des Modells	50
3.2.1	Simulation einer Markov-Kette	50
3.2.2	Random-Walks	53
3.2.3	Das Ehrenfest-Modell	58
3.2.4	Markov-Ketten in der Genetik	61

INHALTSVERZEICHNIS

ii

4 Fazit

64

Literaturverzeichnis

65

Abbildungsverzeichnis

2.1	Der Aufbau eines Neurons	32
2.2	Der Verlauf eines Aktionspotentials	33
2.3	Eine VCON-Rückkopplungsschleife	35
2.4	Ein linearer Amplifier	36

Kapitel 0

Einleitung

In der mathematischen Modellierung wird versucht, wesentliche Mechanismen biologischer Phänomene und deren Parameter zu erfassen und diese in berechenbaren Modellen, wie etwa Gleichungssystemen nutzbar zu machen um Voraussagen für die beobachteten Systeme zu treffen. Es gibt dabei verschiedene Herangehensweisen und Methoden. Einerseits sollen biologische Phänomene durch eine mathematische Modellierung beschrieben werden, andererseits können durch bereits bewährte Modelle Klarheit über und Erklärungen für verschiedene Prozesse in der Natur gewonnen werden. Es gibt also den phänomenologischen Ansatz, welcher ausgehend von biologischen Gegebenheiten nach einer passenden Modellierung sucht und den methodischen Ansatz, der von der anderen Seite kommend mit ausgearbeiteten mathematischen Konzepten versucht den realen Phänomenen zu begegnen.

Da in der hier vorliegenden Arbeit verschiedene Aspekte mathematischer Modellierung untersucht werden sollen, werden diese verschiedenen Ansätze auch dargestellt werden. So wird im ersten Kapitel relativ ausführlich die Thematik von Populationprozessen anhand der Theorie gewöhnlicher Differentialgleichungen vorgestellt.

Im zweiten Kapitel wird dann mit der Neurophysiologie ein völlig anderes Gebiet behandelt. Die Vorstellung in diesem Kapitel beschränkt sich dabei auf eine Einführung eines Neuronen-Modells und der Vorführung seiner Wirkungsweise. An dieser exemplarischen Darstellung eines Neuronen-Modells wird ein phänomenologischer Ansatz für eine Modellerstellung vorgeführt.

Das dritte Kapitel widmet sich dann wieder einer methodologischen Herangehensweise an bestimmte biologische Phänomene. Diesmal anhand wahrscheinlichkeitstheoretischer Methoden, genauer mit einem diskreten stochastischen Prozess, den Markov-Ketten.

Ein weiterer wichtiger Aspekt mathematischer Modellierung in der Biologie, welcher in dieser Arbeit behandelt werden soll, ist die hilfreiche Verwendung eines Computeralgebra-Systems bei der Modellerstellung und der Simulation und Berechnung konkreter Beispiele, wodurch die Untersuchung von komplexeren Systemen erst ermöglicht wird.

In dieser Arbeit wird dafür durchgehend das Computeralgebra-System *Mathematica* verwendet. Alle Graphiken, bis auf die gekennzeichneten Abbildungen, und alle Berechnungen wurden mit diesem Hilfsmittel erstellt.

Kapitel 1

Populationsprozesse

In der Biologie ist die Beobachtung zeitabhängiger Entwicklungen von Organismen eine der Hauptmethoden um Erkenntnisse über Eigenschaften der zu untersuchenden Objekte zu gewinnen.

In diesem Kapitel soll exemplarisch das Wachstumsverhalten von Populationen unter verschiedenen Bedingungen mathematisch modelliert werden. Zunächst wird im ersten Abschnitt das Wachstum einer Art betrachtet, welches durch Beeinflussungen des Lebensraums, durch feindliche Eingriffe und innerartliche Beziehungen kontrolliert wird. Im anschließenden zweiten Teil des Kapitels werden Populationsentwicklungen mehrerer Arten im unterschiedlichen Wechselspiel miteinander untersucht.

1.1 Wachstumsprozesse einer Art

Die Umfang einer Population wird naheliegenderweise an der Anzahl ihrer Individuen gemessen, d.h. diese Größe ist eine natürliche Zahl $x \in \mathbb{N}$. Ein anderer Ansatz besteht in der Angabe der Biomasse der Individuen $x \in \mathbb{R}^+$, also als positive reelle Zahl.

Neben dem 'Wertebereich' der Kenngröße unterscheidet man Modelle auch an Hand ihrer 'Zeitmenge' \mathbb{T} . Entweder man betrachtet Veränderungen in diskreten Zeitschritten $\mathbb{T} \subseteq \mathbb{N}$, beispielsweise Generationenfolgen, oder geht zu kontinuierlicher Zeit $\mathbb{T} \subseteq \mathbb{R}^+$ über. Dies kann für große Populationen gerechtfertigt werden, in denen zwar jede Geburt und jeder Todesfall zu einem festen Zeitpunkt stattfindet, jedoch in der Realität der Abstand zwischen jeder Änderung im einzelnen sehr unregelmäßig sein kann. Der Ansatz der kontinuierlichen Zeit kann insofern als gemittelter Prozess aufgefasst werden, in welchem dann vorwiegend qualitative Gesichtspunkte behandelt werden sollen. Als eine adäquate, einfache Theorie erweist sich hierfür die Stabilitätstheorie gewöhnlicher Differentialgleichungen. Obwohl die Analyse diskreter Modelle, wie zum Beispiel das Wachstumsmodell einer Hasenpopulation von Fibonacci, von Interesse sein kann, sollen in dieser Behandlung von Populationsprozessen ausschließlich kontinuierliche Ansätze behandelt werden.

1.1.1 Exponentielles Wachstum

Es wird eine Population von Lebewesen einer Art in dem utopischen Fall betrachtet, dass das Wachstum weder durch eine äußere Begrenzung wie etwa durch den Lebensraum noch durch interne Faktoren wie Konkurrenz beschränkt wird.

Sei im Folgenden $\mathbb{T} = \mathbb{R}^+$ und $x : \mathbb{T} \rightarrow \mathbb{R}^+$ die Abbildung, welche in Abhängigkeit von der Zeit die Größe der Population darstellt. Diese Funktion soll modelliert werden, indem unter einigen Annahmen eine Differentialgleichung aufgestellt wird, welche den Zuwachsmechanismus der Population beschreibt. Zu einem Zeitpunkt $t \in \mathbb{T}$ sei weiterhin $\Delta t = (t + h) - t$ ein kleines Zeitintervall der Länge $h > 0$ und $\Delta x(t) := x(t + h) - x(t)$ die Populationsdifferenz bezüglich Δt . Es werden die folgenden Annahmen getroffen:

- Die Population wachse proportional zu ihrer Größe: $\Delta x(t) \sim x(t)$
- Je größer das Zeitintervall, desto stärker das Wachstum: $\Delta x(t) \sim \Delta t$, d.h. es existiert lokal um t eine lineare Approximation

$$\exists p \in \mathbb{R} : \quad x(t + h) = x(t) + p\Delta t.$$

Beides zusammen ergibt die folgende Relation:

$$\Delta x(t) \sim x(t)\Delta t$$

Sei a die Proportionalitätskonstante, welche die jährliche Zuwachsrate $\frac{\Delta x(t)}{x(t)}$ darstellt und sich aus der Differenz $a = b - c$ ergibt, wobei b die Geburtenrate und c die Sterberate mit $b > 0$ und $c > 0$ ist. So folgt:

- $\Delta x(t) = ax(t)\Delta t$
- $\frac{\Delta x(t)}{\Delta t} = ax(t)$

Betrachtet man nun den Grenzübergang $\Delta t \rightarrow 0$, so dass die Zeitintervalle Δt immer kleiner werden, dann erhält man die Differentialgleichung

$$x'(t) = ax(t). \tag{1.1}$$

Diese Gleichung wird als **Differentialgleichung des unbegrenzten Wachstums** bezeichnet.

Mit einer festen Populationsgröße als Ausgangspunkt zu Beginn des zu modellierenden Zeitraumes t_0 erhält man das Anfangswertproblem

$$x'(t) = ax(t), \quad x(t_0) = x_0 > 0. \tag{1.2}$$

Es handelt sich um eine lineare Differentialgleichung mit konstantem Koeffizienten, deren Lösung sich rechnerisch durch die Methode der Trennung der Variablen ergibt:

$$x(t) = x_0 e^{a(t-t_0)}. \tag{1.3}$$

Für $a > 0$ gilt also $x(t) \rightarrow \infty$ für $t \rightarrow \infty$. Für negatives a ergibt sich $x(t) \rightarrow 0$ für $t \rightarrow \infty$, d.h. ein Aussterben der Population in exponentieller Geschwindigkeit, unabhängig von der Größe des Anfangswertes $x_0 > 0$.

Um die Lösbarkeit auch aller folgenden Differentialgleichungen (DGLen) zu gewährleisten, sei hier der Satz von Picard-Lindelöf aus [AUL] zitiert.

Satz 1.1 (Picard-Lindelöf)

Sei D ein offenes, beschränktes Gebiet in \mathbb{R}^n , $n \in \mathbb{N}$ und $f : \mathbb{R} \times D \rightarrow \mathbb{R}^n$ eine stetige und bezüglich x Lipschitzstetige Funktion. Dann gilt:

Für jedes Anfangswertproblem der Form

$$\frac{dx}{dt} = f(t, x), \quad x(t_0) = x_0$$

mit (t_0, x_0) aus D existiert stets eine eindeutig bestimmte Lösung auf D .

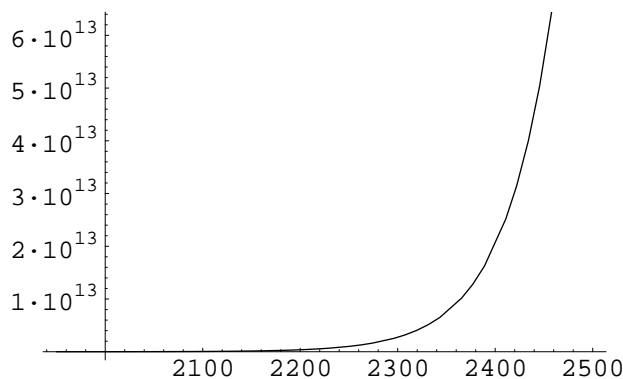
Für die hier auftretenden DGLen werden die Voraussetzungen des Satzes immer gegeben sein, da es sich bei den rechten Seiten stets um Polynome in x und endlichen Grades handelt.

Mit *Mathematica* lässt sich das Anfangswertproblem (1.2) bei geeigneten Werten mit dem Befehl `DSolve` leicht lösen und veranschaulichen.

Die folgenden Daten entsprechen einer Schätzung der Weltbevölkerung im Jahre 1950, siehe [KOE]. Für eine Proportionalitätskonstante $a = 0,02$ kann dann eine Vorhersage für das Bevölkerungswachstum in den nächsten Jahren berechnet werden:

```
In[1]:= sol = DSolve[{x'[t] == a x[t], x[t0] == x0}, x[t], t]
         X = x[t] /. sol[[1]] /. {x0 -> 2.5 10^9, t0 -> 1950, a -> 0.02};
         Plot[X, {t, 1950, 2500}]
```

```
Out[1]= {{x[t] -> e^{a t - a t0} x0}}
```



```
Out[3]= - Graphics -
```

Mit diesem Modell zeigt sich ein enormes Wachstum der Population, falls die Population zum Zeitpunkt t_0 bereits die Größe $2,510^9$ hat und man von einer Proportionalitätskonstante $a = 0,02$ ausgeht. Die Vorhersage des Modells für das Anwachsen der Bevölkerung ist fatal.

In[4]:= $\mathbf{x} / . \mathbf{t} \rightarrow 2500$

Out[4]= 1.49685×10^{14}

Dies bedeutet ein so starkes Ansteigen der Weltbevölkerung, dass der Lebensraum, die Erdoberfläche, pro Person nur noch wenige Quadratmeter betragen würde.

Natürlich ist das Modell sensitiv von den gewählten Parametern abhängig. Das Anfangswertepaar (t_0, x_0) geht selbstverständlich aus empirischen Messungen hervor, die Proportionalitätskonstante allerdings muss, um realistisch zu sein, mit einem speziellen, aus der Statistik kommenden Verfahren, der *Methode der kleinsten Quadrate* geschätzt werden.

Parameterschätzung: Regression - Methode der kleinsten Quadrate

Ausgangspunkt ist ein Datensatz (t_i, x_i) , $i = 1, \dots, n$, hinter dem ein deterministischer Zusammenhang vermutet wird. Nun werden $s \leq n$ Ansatzfunktionen f_1, \dots, f_s gewählt, die die vermutete Beziehung möglichst gut repräsentieren, Ziel ist nun geeignete Koeffizienten $\gamma_1, \dots, \gamma_s$ zu gewinnen, so dass die Linearkombination $\gamma_1 f_1(t) + \dots + \gamma_s f_s(t)$ sich im quadratischen Mittel möglichst optimal den Daten anpasst. D.h. gesucht werden $\gamma_1, \dots, \gamma_s$ mit der Eigenschaft

$$\sum_{k=1}^n \left(x_k - \sum_{i=1}^s f_i(t_k) \gamma_i \right)^2 \stackrel{!}{=} \min.$$

Dies ist ein klassisches Extremwertproblem, welches auf Grund des vorkommenden Quadrats stets ein globales Minimum besitzt. Setzt man den Gradienten gleich Null,

$$\nabla_{\gamma} \left(\sum_{k=1}^n \left(x_k - \sum_{i=1}^s f_i(t_k) \gamma_i \right)^2 \right) \stackrel{!}{=} 0,$$

so erhält man ein lineares Gleichungssystem, das so genannte *Normalgleichungssystem*. Schreibt man

$$A := \begin{pmatrix} f_1(t_1) & \dots & f_s(t_1) \\ \vdots & & \vdots \\ f_1(t_n) & \dots & f_s(t_n) \end{pmatrix},$$

so ist der Ansatz in Vektorschreibweise

$$(x - A\gamma, x - A\gamma) \stackrel{!}{=} \min.$$

A muß hier so gewählt werden, dass der Rang voll ist. Dann ist das Bild

$$L := \{A\gamma : \gamma \in \mathbb{R}^s\}$$

wie aus der Linearen Algebra bekannt, ein s -dimensionaler Untervektorraum von \mathbb{R}^n . Sei Π_L die Projektion von \mathbb{R}^n auf L , dann soll gelten,

$$\Pi_L x = A\gamma$$

Mit Methoden der Linearen Algebra [GEO] kann gezeigt werden, dass

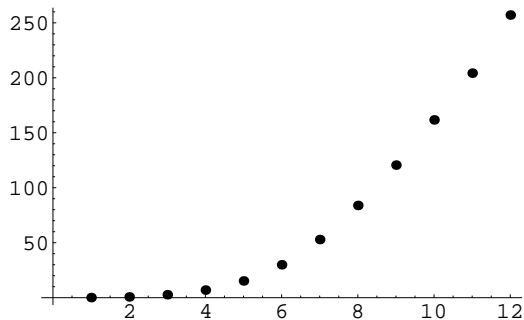
$$\hat{\gamma} := (A^T A)^{-1} A^T x \quad (1.4)$$

die einzige Lösung hierfür ist und das Normalgleichungssystem erfüllt. (1.4) minimiert damit die Summe der quadratischen Fehler.

Als Beispiel innerhalb dieses Einschubs zur Parameterbestimmung, seien hier die offiziellen Anzahlen der Aids-Infizierten in den Vereinigten Staaten im Zeitraum von 1981 bis 1992 in Tausend genannt [YSH], welche approximiert werden sollen.

```
In[1]:= {{1, .097}, {2, .709}, {3, 2.698}, {4, 6.928},
         {5, 15.242}, {6, 29.944}, {7, 52.902}, {8, 83.903}, {9, 120.612},
         {10, 161.711}, {11, 204.247}, {12, 257.085}} // MatrixForm
p1 = ListPlot[{{1, .097}, {2, .709}, {3, 2.698}, {4, 6.928}, {5, 15.242},
              {6, 29.944}, {7, 52.902}, {8, 83.903}, {9, 120.612}, {10, 161.711},
              {11, 204.247}, {12, 257.085}}, PlotStyle -> PointSize[0.02]]
```

```
Out[1]//MatrixForm=
```

$$\begin{pmatrix} 1 & 0.097 \\ 2 & 0.709 \\ 3 & 2.698 \\ 4 & 6.928 \\ 5 & 15.242 \\ 6 & 29.944 \\ 7 & 52.902 \\ 8 & 83.903 \\ 9 & 120.612 \\ 10 & 161.711 \\ 11 & 204.247 \\ 12 & 257.085 \end{pmatrix}$$


Mit der Vermutung eines schwach exponentiellen Wachstums werden passende Ansatzfunktionen gesucht. Es zeigt sich nach einigen Versuchen, dass die Funktionen $f_1(t) = 1$, $f_2(t) = e^{0,02t}$, $f_3(t) = e^{0,1t}$ den geforderten Ansprüchen entsprechen und die Daten gut approximieren, wie die nachfolgende Darstellung zeigt.

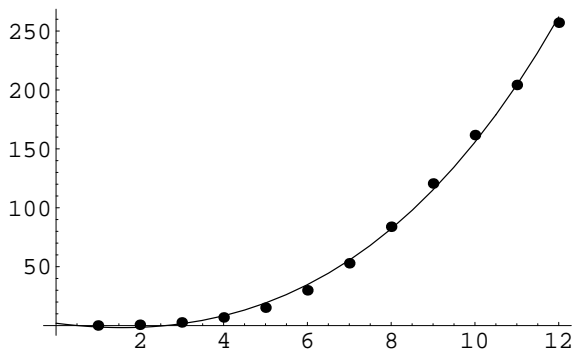
```
In[3]:= Clear[f1, f2, f3, p2, sol, F, X, A]
f1[t_] := 1 // N;
f2[t_] := Exp[0.02 t] // N;
f3[t_] := Exp[0.1 t] // N;
```

$$A = \begin{pmatrix} f1[1] & f2[1] & f3[1] \\ f1[2] & f2[2] & f3[2] \\ f1[3] & f2[3] & f3[3] \\ f1[4] & f2[4] & f3[4] \\ f1[5] & f2[5] & f3[5] \\ f1[6] & f2[6] & f3[6] \\ f1[7] & f2[7] & f3[7] \\ f1[8] & f2[8] & f3[8] \\ f1[9] & f2[9] & f3[9] \\ f1[10] & f2[10] & f3[10] \\ f1[11] & f2[11] & f3[11] \\ f1[12] & f2[12] & f3[12] \end{pmatrix};$$

```
In[8]:= werte := {.097, .709, 2.698, 6.928, 15.242,
29.944, 52.902, 83.903, 120.612, 161.711, 204.247, 257.085};
F = Inverse[(Transpose[A].A)].Transpose[A];
LinKoeff = F.werte;
F.werte // MatrixForm
p2 = Plot[
LinKoeff[[1]] f1[t] + LinKoeff[[2]] f2[t] + LinKoeff[[3]] f3[t], {t, 0, 12}];
Show[p1, p2];
```

```
Out[11]//MatrixForm=
```

$$\begin{pmatrix} 1561.32 \\ -1892.65 \\ 333.329 \end{pmatrix}$$



Wichtig für biologische Modelle ist oft der Fall linearer Ansatzfunktionen, das bedeutet, dass eine Gerade der Form $x = \gamma_1 + t\gamma_2$ wird gesucht, welche die Datenwerte am besten approximiert, d.h. am wenigsten von den empirische Daten abweicht. In diesem Fall können sehr einfache Formeln ohne das numerische Problem der Matrizeninvertierung gewonnen werden.

Eine kurze Skizze der Herleitung [YSH]:

Fehler pro Datenpaar:

$$e_i = x_i - (\gamma_2 t_i + \gamma_1), i = 1, \dots, n .$$

Minimierung des quadratischen Fehlers:

$$E(a, m) = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (x_i - (\gamma_2 t_i + \gamma_1))^2 \quad (1.5)$$

Die Kurvendiskussion für E nach γ_1 und γ_2 :

$$0 = \frac{dE}{d\gamma_2} = -2 \sum_{i=1}^n t_i (x_i - (\gamma_2 t_i + \gamma_1)) \quad (1.6)$$

$$0 = \frac{dE}{d\gamma_1} = -2 \sum_{i=1}^n x_i - (\gamma_2 t_i + \gamma_1) . \quad (1.7)$$

Die beiden Gleichungen (1.6) und (1.7) werden vereinfacht zu

$$0 = \sum_{i=1}^n t_i x_i - a \sum_{i=1}^n t_i^2 - \gamma_1 \sum_{i=1}^n t_i , \quad (1.8)$$

$$0 = \sum_{i=1}^n x_i - a \sum_{i=1}^n t_i - k\gamma_1 . \quad (1.9)$$

Auflösen nach γ_2 und γ_1 liefert schließlich:

$$\gamma_2 = \frac{n \sum_{i=1}^n t_i x_i - (\sum_{i=1}^n t_i)(\sum_{i=1}^n x_i)}{n \sum_{i=1}^n t_i^2 - (\sum_{i=1}^n t_i)^2} \quad (1.10)$$

$$\gamma_1 = \frac{(\sum_{i=1}^n t_i^2)(\sum_{i=1}^n x_i) - (\sum_{i=1}^n t_i)(\sum_{i=1}^n t_i x_i)}{n \sum_{i=1}^n t_i^2 - (\sum_{i=1}^n t_i)^2} \quad (1.11)$$

Somit kann die Regressionsgerade $x = \gamma_2 t + \gamma_1$ berechnet werden und außerdem die Wachstumsrate $a = \gamma_2$ geschätzt werden.

In *Mathematica* sind bereits Funktionen implementiert, welche die Methode der kleinsten Quadrate verwenden und deshalb eine leichte Anwendung des Verfahrens ermöglichen. So führt die Funktion `Fit` sowohl Berechnungen von Regressionsgeraden, aber auch Approximationen von Daten mit schwierigeren Zusammenhängen durch. Es müssen neben dem Datensatz lediglich die Ansatzfunktionen f_1, \dots, f_s der Funktion `Fit` übergeben werden.

Wieder zurückgekehrt zum Modell des unbegrenzten Wachstums einer Population, liefert die Lösung (1.3) der DGL einen exponentiellen Zusammenhang. Die Idee ist nun, statt wie im Beispiel der Aids-Infizierten eine Linearkombination von Exponentialfunktionen zu wählen, dass die Daten logarithmiert werden und für diese Daten dann ein linearer Zusammenhang geschätzt wird. Die gefundene Regressionsgerade wird dann anschließend rückexponentiert.

(1.5) nimmt also die Gestalt

$$E(\gamma_2, \gamma_1) = \sum_{i=1}^n (\ln(x_i) - (\gamma_2 t_i + \ln(x_0)))^2 \quad (1.12)$$

an. Als Achsenabschnitt γ_1 der Ansatzgeraden wird nämlich gleich der Anfangswert x_0 der Population gesetzt. Nach Ableiten von (1.12) nach γ_2 , Nullsetzen und schließlich Auflösen nach γ_2 erhält man so

$$a = \gamma_2 = \frac{\sum_{i=1}^n t_i (\ln(x_i) - \ln(x_0))}{\sum_{i=1}^n t_i^2}. \quad (1.13)$$

Die Proportionalitätskonstante a konnte also bestimmt werden.

Damit wurden die verschiedenen Möglichkeiten der Parameterschätzung mit der Methode der kleinsten Quadrate vorgestellt und für die folgenden Anwendungen ist sichergestellt, dass die auftretenden Parameter in sinnvoller Weise ermittelt werden können.

1.1.2 Logistisches Wachstum

Für das Beispiel des Populationswachstums gehen wir nun von der natürlichen Begrenzung des Lebensraumes für die Population aus. Diese ergibt sich aus der Beschränkung der zur Verfügung stehenden Ressourcen wie Nahrung und Raum, aber auch durch innerartliche Konkurrenz. Das Modell zur Beschreibung der Populationsgröße ändert sich also durch die Annahme, dass das Ausmaß der Population nicht nur das Wachstum fördert, sondern auch beschränkt, da jedes Individuum in Konkurrenz um den begrenzten Lebensraum mit den anderen Populationsmitgliedern steht.

Zu den bisherigen Annahmen des ungehinderten Wachstums einer Art muss nun auch der Aspekt der Größe der Population als Nachteil für das einzelne Individuum berücksichtigt werden. Da die Mitglieder der Population untereinander konkurrieren, nimmt man an, die Reduktion der Population durch innerartlichen Wettbewerb sei proportional zu $x(t)^2$. Mit beiden Prämissen zusammen erhält man die **Differentialgleichung des logistischen Wachstums**

$$x'(t) = ax(t) - bx(t)^2, \quad (1.14)$$

wobei $a > 0$ wieder die Zuwachsrate und $b > 0$ eine Proportionalitätskonstante für die innerartliche Konkurrenz darstellt. Auch diese DGL lässt sich durch die Methode der Trennung der Variablen lösen, nachdem wieder eine Vorgabe über die Anfangsgröße der Population die Differentialgleichung zu einem Anfangswertproblem vervollständigt:

$$\frac{dx}{dt} = ax(t) - bx(t)^2 \quad x(t_0) = x_0 \quad (1.15)$$

Trennung der Variablen:

$$\frac{dx}{ax - bx^2} = dt \quad (1.16)$$

$$\int_{x_0}^x \frac{dr}{ar - br^2} = \int_{t_0}^t ds = t - t_0 \quad (1.17)$$

Um die linke Seite zu integrieren ist eine Partialbruchzerlegung hilfreich:

$$\begin{aligned}\frac{1}{ar - br^2} &= \frac{1}{r(a - br)}, \\ \frac{A}{r} + \frac{B}{a - br} &= \frac{A(a - br) + Br}{r(a - br)} = \frac{Aa + (B - bA)r}{r(a - br)}\end{aligned}$$

Durch Koeffizientenvergleich ergibt sich das Gleichungssystem

$$\begin{aligned}Aa &= 1 \\ B - bA &= 0\end{aligned}$$

Somit gilt $A = \frac{1}{a}$ und $B = \frac{b}{a}$. Nun kann leicht eine geschlossene Form der Stammfunktion auf der linken Seite gefunden werden.

$$\begin{aligned}\int_{x_0}^x \frac{1}{r(a - br)} \, dr &= \frac{1}{a} \int_{x_0}^x \frac{1}{r} + \frac{b}{a - br} \, dr \\ &= \frac{1}{a} \left(\ln \left(\frac{x}{x_0} \right) + \ln \left(\left| \frac{a - bx_0}{a - bx} \right| \right) \right) \\ &= \frac{1}{a} \ln \left(\frac{x}{x_0} \left| \frac{a - bx_0}{a - bx} \right| \right)\end{aligned}$$

Also hat die ursprüngliche Gleichung (1.17) die Form,

$$a(t - t_0) = \ln \left(\frac{x}{x_0} \frac{a - bx_0}{a - bx} \right), \quad (1.18)$$

wenn man zeigen kann, dass $\frac{a - bx_0}{a - bx} > 0, \forall t \in \mathbb{R}$. Dazu muss eine kurze Fallunterscheidung vorgenommen werden:

Beweis: Zunächst wird festgesetzt, dass $a, b > 0$ gilt.

Es werden dann die verschiedenen Fälle von x_0 untersucht. Dabei werden die beiden Fälle $x_0 = 0$ und $x_0 = \frac{a}{b}$ extra behandelt, da dies die Nullstellen der rechten Seite von (1.14) sind. D.h. hier gilt $x_0 = 0 \Rightarrow x = 0$, Analoges gilt für $\frac{a}{b}$. In diesen Fällen ist (1.18) legitim.

1. Fall: $0 < x_0 < \frac{a}{b}$

Sei $x_0 := x(0) < \frac{a}{b}$. Es gilt dann für (1.14): $x'(0) = ax_0 - bx_0^2 = x_0(a - bx_0) > 0$ und somit $a - bx_0 > 0$.

Da x nach Definition stetig differenzierbar ist, gilt: Es existiert eine Umgebung $U \subset \mathbb{R}$ um Null, so dass für alle $t \in U$ $0 < x(t)(a - bx(t)) = x'(t)$ gilt. Insbesondere gilt dann auch $a - bx(t) > 0$ und $x(t) < \frac{a}{b}$.

In der Argumentation wurde aber nicht verwendet, dass diese Umgebung nur um den Nullpunkt existieren kann. Definiert man also allgemein $x_0 := x_{t_0}$ für ein beliebiges $t_0 \in \mathbb{R}$ mit $0 < x_{t_0} < \frac{a}{b}$, so erhält man ebenfalls $0 < x(t)(a - bx(t)) = x'(t)$ und somit $x(t) < \frac{a}{b}$ für alle $t \in U$, mit einer entsprechenden Umgebung U .

Damit wurde gezeigt, dass für $0 < x_0 < \frac{a}{b}$ auch $0 < x < \frac{a}{b}$ gilt, und außerdem für alle t auch $x'(t) > 0$ gegeben ist. x ist in diesem Fall also monoton steigend. Dies bedeutet

aber, dass dann $a - bx_0 > 0$ und auch $a - bx > 0$ gilt. In diesem Fall ist $\frac{a-bx_0}{a-bx}$ also stets positiv und (1.18) gilt.

2.Fall: $x_0 > \frac{a}{b}$

Mit der analogen Argumentationsweise kann man zeigen, dass in diesem Fall für alle $t \in \mathbb{R}$ $x(t) > \frac{a}{b}$ gilt, und schließlich, dass dann $\frac{a-bx_0}{a-bx}$ auch immer positiv ist. Somit kann der Betrag weggelassen werden. \square

Anwendung der Exponentialfunktion auf beiden Seiten von (1.18) liefert

$$e^{a(t-t_0)} = \left(\frac{x}{x_0}\right) \left(\frac{a-bx_0}{a-bx}\right).$$

Nun soll nach $x(t)$ aufgelöst werden:

$$(a - bx_0 + bx_0 e^{a(t-t_0)})x = ax_0 e^{a(t-t_0)}$$

Man erhält schließlich:

$$x(t) = \frac{ax_0}{bx_0 + (a - bx_0)e^{-a(t-t_0)}} \quad (1.19)$$

Die Funktion, die wir hiermit gefunden haben wird als **logistische Funktion** bezeichnet.

Definition 1.2 Eine DGL der Form $\frac{dx}{dt} = f(x)$ heißt **autonom**, d.h. ihre rechte Seite hängt nicht explizit von t ab.

Autonome DGLen haben besonders schöne Eigenschaften, insbesondere sind ihre Lösungen translationsinvariant, d.h. für Lösungen x mit $x(0) = x_1$ und y mit $y(t_0) = x_1$ gilt

$$\forall t \in \mathbb{R} : \quad x(t - t_0) = y(t).$$

Definition 1.3 Ein Wert $x^* \in \mathbb{R}^d$, $d \in \mathbb{N}$ heißt **Gleichgewichtspunkt** (Ruhelage, Equilibrium) für eine DGL $\frac{dx}{dt} = f(t, x(t))$, falls gilt

$$f(t, x^*) = 0 \quad \forall t \in \mathbb{R}.$$

Anders ausgedrückt sind Gleichgewichtsstellen Punkte, an denen sich die Funktion x lokal nicht ändert, also lokale Extrema.

Für die logistische Funktion ergeben sich wegen $x'(t) = x(t)(a - bx(t))$ die beiden Gleichgewichte $x^+ = 0$ und $x^* = \frac{a}{b}$.

Definition 1.4 Ein Gleichgewichtspunkt x^* für eine DGL $\frac{dx}{dt} = f(t, x(t))$, heißt **stabil**, wenn es Umgebungen U und V von x^* gibt, so dass für alle Anfangswerte in U die Lösungen x mit $x(0) \in U$ gilt,

$$x(t) \in V \quad \forall t \in \mathbb{R}.$$

Mit anderen Worten, alle Lösungen mit Anfangswerten in U bleiben innerhalb einer Umgebung V .

Definition 1.5 Ein Gleichgewichtspunkt x^* für eine DGL $\frac{dx}{dt} = f(t, x(t))$, heißt **attraktiv**, falls es eine Umgebung U von x^* gibt, so dass gilt

$$x(0) \in U \implies \lim_{t \rightarrow \infty} x(t) = x^*.$$

Der Gleichgewichtspunkt x^* heißt **asymptotisch stabil**, falls er stabil und attraktiv ist.

Diese Definitionen sind rein qualitativer Art und kaum direkt überprüfbar. Allerdings gibt es für autonome DGLen mit stetig differenzierbarer rechten Seite mittels Taylorentwicklung um den Gleichgewichtspunkt ein hinreichendes Kriterium für asymptotische Stabilität [AUL].

Satz 1.6 (Kriterium für asymptotische Stabilität autonomer DGLen)

Ein Gleichgewichtspunkt x^* einer autonomen DGL $\frac{dx}{dt} = f(x)$ mit $f(x) \in C^1$ ist asymptotisch stabil, wenn die Jacobi-Matrix an der Stelle x^* nur Eigenwerte mit negativen Realteilen besitzt.

Betrachtet man die Linearisierungen um die beiden Gleichgewichte der logistischen Funktion, so ist mit dieser Methode leicht zu erkennen, dass $x^* = \frac{a}{b}$ der asymptotisch stabile und $x^+ = 0$ der instabile Gleichgewichtspunkt ist, denn es gilt für die Jacobi-Matrix der rechten Seite, $f(x)$:

$$\begin{aligned} \frac{df}{dx}(x^+) &= a - 2bx^+ = a > 0 \\ \frac{df}{dx}(x^*) &= a - 2bx^* = a - 2b\left(\frac{a}{b}\right) = -a < 0 \end{aligned}$$

Über die Grenzwertbetrachtung der expliziten Lösung (1.19) für $t \rightarrow \infty$

$$\lim_{t \rightarrow \infty} x(t) = \frac{ax_0}{bx_0 + (a - bx_0)e^{-a(t-t_0)}} = \frac{a}{b} = x^*, \quad x_0 \neq x^+$$

erhält man, dass die Umgebung U von x^* der asymptotisch stabilen Anfangswerte gleich ganz $\mathbb{R} \setminus \{x^+\}$ bildet.

So sieht man, dass das logistische Wachstum einer Population gegen seine nichttriviale, stabile Gleichgewichtsstelle konvergiert, die so genannte **Grenzpopulation**. Für Anfangspopulationsgrößen x_0 mit $x^+ < x_0 < x^*$ konvergiert die Lösung (1.19) von unten gegen x^* . Dagegen nähert sich die Lösungsfunktion $x(t)$ für Anfangswerte x_0 mit $x^* < x_0$ von oben an das stabile Gleichgewicht an.

Mit *Mathematica* wird nun zunächst die logistische Funktion durch ein Anwenden der Funktion `DSolve` ermittelt und an Hand von gewählten Werten, welche in Form von Regelwerken in die symbolische Lösungsfunktion eingesetzt werden, wird ein konkretes Beispiel in einer Graphik veranschaulicht.

Es handelt sich hierbei wieder um eine, dieses Mal etwas realistischere Modellierung der Weltbevölkerung [KOE]. Das Anfangswertepaar ist wieder $(t_0, x_0) = (1950, 2,5 \cdot 10^9)$.

Aus empirischen Messungen wurde als Wachstumsrate $a = 0,025$ ermittelt. Außerdem ist für dieses Beispiel besonders, dass der Wert $x'(t) = 0,02$ bekannt ist, da angenommen wird, dass die Bevölkerung bei einer Größe von 2.5 Milliarden um 2 % jährlich ansteigt. Aus diesen Vorgaben kann dann der Parameter b bestimmt werden.

Abgesehen von diesem Beispiel muß normalerweise für die Berechnung von b eine Schätzung der Grenzpopulation $x = \frac{a}{b}$ vorgenommen werden und da die Wachstumsrate a stets empirisch durch die Methode der kleinsten Quadrate ermittelt werden kann, ist die Berechnung von b damit möglich.

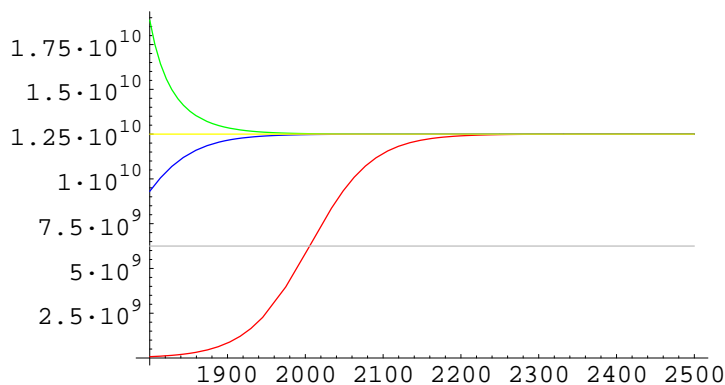
```
In[1]:= Needs["Graphics`Colors`"]
```

```
In[2]:= logistisch = DSolve[{x'[t] == a x[t] - b x[t]^2, x[t0] == x0}, x[t], t]
```

```
Out[2]= {{x[t] -> \frac{a e^{a t} x_0}{a e^{a t_0} + b e^{a t} x_0 - b e^{a t_0} x_0}}}
```

```
In[3]:= Clear[regel1, regel2, regel3, a, b, x0, t0]
regel1 = {x0 -> 2.5 10^9, t0 -> 1950, a -> 0.025};
B = Solve[0.02 == a - b x0, b] /. regel1;
regel1 = AppendTo[regel1, B[[1]][[1]]];
Gleichgewicht = a / b /. regel1;
Wendepunkt = a / (2 b) /. regel1;
regel2 = Drop[regel1, 1];
regel2 = PrependTo[regel2, x0 -> Gleichgewicht + 10^8];
regel3 = Drop[regel1, 1];
regel3 = PrependTo[regel3, x0 -> Gleichgewicht - 10^8];
```

```
In[13]:= X1 = x[t] /. logistisch[[1]] /. regel1;
X2 = x[t] /. logistisch[[1]] /. regel2;
X3 = x[t] /. logistisch[[1]] /. regel3;
Plot[{X1, X2, X3, Gleichgewicht, Wendepunkt}, {t, 1800, 2500},
PlotRange -> All, PlotStyle -> {Red, Green, Blue, Yellow, Gray}]
```



```
Out[16]= - Graphics -
```

In der obigen Abbildung ist gut die Attraktivität des asymptotisch stabilen Gleichgewichtspunkt $x^* = \frac{a}{b}$ (gelbe Linie) zu erkennen, denn unabhängig von der Anfangsgröße pendelt sich die Populationsgröße (rote, blaue, grüne Linie) in diesen Wert ein. Außerdem wurde bereits der Wendepunkt der logistischen Funktion grau miteingezeichnet.

Dieser berechnet sich als Nullstelle der zweiten Ableitung der Funktion (1.19).

$$\begin{aligned} x''(t) &= \frac{d}{dt}(ax(t) - bx(t)^2) \\ &= (a - 2bx(t))x'(t) \\ &= (a - 2bx(t))(ax(t) - bx(t)^2). \end{aligned}$$

$$0 = (a - 2bx(t))x(t)(a - bx(t))$$

Daraus folgt:

$$x(t) = \frac{a}{2b}, \quad x(t) = 0, \quad x(t) = \frac{a}{b}.$$

Also ist $x(t) = \frac{a}{2b}$ der Wendepunkt der logistischen Funktion. Anhand dieses Punktes läßt sich das Wachstumsverhalten genauer beschreiben:

Im Falle $x^+ < x_0 < x^*$ gilt, $x(t)$ wächst konvex, bis die Hälfte der Grenzpopulation erreicht ist, danach ist das Wachstum konkav bis zur Erreichung des Gleichgewichts. In diesem Fall lassen sich die Schranken für die logistische Funktion für alle $t \in \mathbb{R}$ angeben:

$$0 < \frac{ax_0}{(bx_0 + (a - bx_0)e^{-a(t-t_0)})} < \frac{a}{b}$$

1.1.3 Wachstum einer Art bei regelmäßigen Dezimierungseingriffen

Eine Population einer Art ist in ihrem Lebensraum selten allein auf sich gestellt, sondern unterliegt den Einflüssen von diversen umweltbedingten Dezimierungsfaktoren, regelmäßigen Eingriffen anderer Arten.

So ist zum Beispiel für die Landwirtschaft oder die Ökologie von Interesse, welche Anteile einer Population geerntet, oder einfach entfernt werden können, ohne den Gesamtbestand der Art zu stark zu beeinträchtigen und deren Fortbestehen zu gewährleisten. Die Ausgangssituation ist ein logistisches Wachstum einer Art. Wird nun regelmäßig der Anteil f mit $f > 0$ der Population, also $f x(t)$ als Ernteertrag abgezogen, so ändert sich die Differentialgleichung (1.14) zu

$$x'(t) = ax(t) - bx(t)^2 - fx(t). \quad (1.20)$$

Natürlich bedeutet dies quantitativ nur eine verringerte Wachstumsrate $a - f$. Mit diesem Wert pendelt sich die Populationsgröße aber auch auf einen niedrigeren Stand der Grenzpopulation $x^* = \frac{a-f}{b}$ ein. Das bedeutet dann langfristig einen Ernteertrag von $fx^* = \frac{f(a-f)}{b}$. Wird außerdem f zu groß gewählt, so dass $f > a$, so stirbt die Population unter diesem Einfluß langfristig aus.

Wird der Ernteertrag allerdings unabhängig von der jeweiligen Größe der Population, stets in einem festen Ausmaß y_0 mit $y_0 > 0$ geerntet, so sieht das Wachstumsmodell folgendermaßen aus:

$$x'(t) = ax(t) - bx(t)^2 - y_0 \quad (1.21)$$

Die Gleichgewichte der Lösungsfunktion des Modells werden mit *Mathematica* berechnet und in die eindimensionale Jacobi-Matrix eingesetzt.

```
In[1]:= k[x_] := a x - b x^2 - y0
        s = Solve[a x - b x^2 - y0 == 0, x]
        D[k[x], x] /. s[[1]]
        D[k[x], x] /. s[[2]]
```

```
Out[2]= {{x -> -(-a - Sqrt[a^2 - 4 b y0]) / (2 b)}, {x -> -(-a + Sqrt[a^2 - 4 b y0]) / (2 b)}}
```

```
Out[3]= -Sqrt[a^2 - 4 b y0]
```

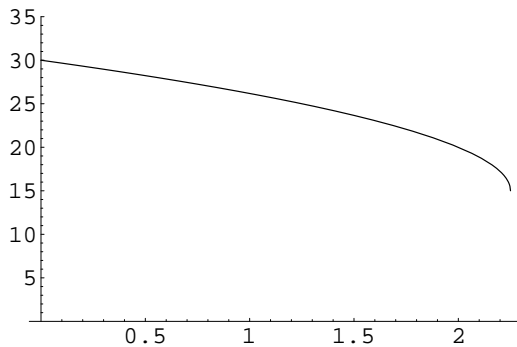
```
Out[4]= Sqrt[a^2 - 4 b y0]
```

Es gilt $x^+ = -\frac{-a - \sqrt{a^2 - 4by_0}}{2b}$ und $x^* = -\frac{-a + \sqrt{a^2 - 4by_0}}{2b}$, wobei wegen der Stabilitätsberechnung der Erste der stabile der beiden ist, da dessen entsprechender einziger Eigenwert einen negativen Realteil hat.

Seien im Folgenden die Werte mit $a = 0,3$ und $b = 0,01$ exemplarisch fixiert. Wir betrachten nun das Stabilitätsverhalten bei wachsendem y_0 . Auf Grund der Form des stabilen Gleichgewichtspunktes existiert nur dann eine Lösung, wenn die Diskriminante positiv ist. Wenn also y_0 den Wert $\frac{a^2}{4b}$ überschreitet, so existiert kein Gleichgewichtspunkt mehr. Davor hat er als Funktion in y_0 die Form:

```
In[8]:=
```

```
Plot[-(a - Sqrt[a^2 - 4 b y0]) / (2 b), {y0, 0, (a^2) / (4 b)}, PlotRange -> {0, 35} /.
      {a -> 0.3, b -> 0.01};
```



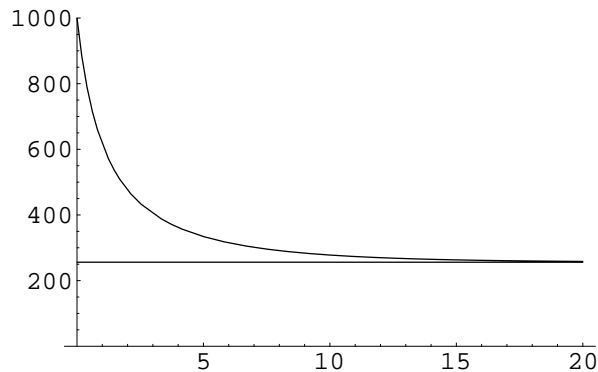
Um sich ein konkretes Beispiel mit *Mathematica* berechnen zu lassen, muss man eine numerische Berechnung der DGL mit der *Mathematica*-Funktion `NDSolve` vornehmen. Bei einer Anfangsgröße von $x(0) = 1000$ und einer Ernterate $y_0 \leq \frac{a^2}{4b}$ kann man dann sehen, wie sich das Verhalten einer Population der auf lange Sicht entwickelt: Die Populationsgröße pendelt sich in den stabilen Gleichgewichtswert ein.

```

In[1]:= a = 0.3;
        b = 0.001;
        y0 = a^2 / (8 b)
        sol = NDSolve[{x'[t] == a x[t] - b (x[t])^2 - y0, x[0] == 1000}, x[t], {t, 0, 50}];
        P1 = Plot[Evaluate[x[t] /. sol], {t, 0, 20}, PlotRange -> {0, 1000}]
        P2 = Plot[-(a - Sqrt[a^2 - 4 b y0]) / (2 b), {t, 0, 20}, PlotRange -> {0, 1000}];
        Show[P1, P2]

```

Out[3]= 11.25



1.2 Populationsmodelle für zwei Arten in einem Lebensraum

Bisher wurden nur die Wachstumsprozesse einer einzigen Spezies, beeinflusst durch umweltbedingte Begrenzungsfaktoren wie die Knappheit der Ressourcen, aber auch die Situation einer Art als Beute für andere Arten im vorangegangenen Abschnitt bereits behandelt.

Nun soll das Wachstumsverhalten von Populationen mehrerer Arten, welche untereinander in Beziehung stehen, mit in die Modellierung einbezogen werden.

In diesem Unterkapitel sollen jeweils zwei Arten mit unterschiedlichen Beziehungen berücksichtigt werden. Die verschiedenen Modelltypen werden je nach der Art der zwischenartlichen Beziehung klassifiziert. Hier sollen neben einer Vorstellung von Räuber-Beute-Systemen auch Modelle konkurrierender Arten und Symbiosen behandelt werden.

1.2.1 Räuber-Beute-Modelle

Im folgenden Unterabschnitt sei nun das Szenario von zwei verschiedenen Arten gegeben, welche in einer so genannten *Räuber-Beute-Beziehung* stehen, das heißt, die Art der Räuber ist auf die Beute-Art als ein Hauptnahrungsmittel angewiesen. Sei $x : \mathbb{R} \rightarrow \mathbb{R}^+$ die Funktion, die die Größe der Beutepopulation in Biomasse und in Abhängigkeit der Zeit darstellt. Sei entsprechend $y : \mathbb{R} \rightarrow \mathbb{R}^+$ für die Räuberpopulation definiert.

Räuber-Beute-Modelle ohne externe Regulatoren

Es wird erst einmal die Annahme getroffen, dass die Entwicklung der Beutepopulation auf Grund von unbeschränkten Ressourcen nur von der internen Geburten- und Sterberate, zusammengefasst im Parameter Wachstumsrate $a > 0$, abhängt.

$$\frac{dx}{dt} = ax(t).$$

Für die Räuberpopulation jedoch ist die Reproduktionsrate auch durch die Größe der Beutepopulation geregelt. Gibt es keine Beutetiere, schrumpft die Räuberpopulation mit der Zeit. $c > 0$ stellt die Sterberate der Räuber dar.

$$\frac{dy}{dt} = -cy(t).$$

Die Wechselwirkung der Räuber-Beute-Beziehung spiegelt sich natürlich auch in der Beutepopulation wieder. Man nimmt an, die Reduktion der Beutepopulation ist proportional von $x(t)$ und $y(t)$ abhängig und wird durch $b > 0$, die Sterberate durch räuberische Angriffe dargestellt.

$$\frac{dx}{dt} = ax(t) - bx(t)y(t).$$

Andererseits ist das Räuberwachstum ebenfalls proportional von den Größen $x(t)$ und $y(t)$ beeinflusst und zwar mit der Vermehrungsrate durch Beuteverzehr d . Es gelte auch hierfür im Folgenden $d > 0$.

$$\frac{dy}{dt} = -cy(t) + dx(t)y(t).$$

Die beiden DGLen bilden zusammen das Differentialgleichungssystem

$$\frac{dx}{dt} = ax(t) - bx(t)y(t) \tag{1.22}$$

$$\frac{dy}{dt} = -cy(t) + dx(t)y(t) \tag{1.23}$$

Für das Lösen des Systems wird dann ein Anfangswert $(t_0; x_0, y_0)$ benötigt.

Die Mathematiker Lotka und Volterra stellten zu Beginn des 20. Jahrhunderts unabhängig voneinander erstmals dieses Differentialgleichungssystem auf. Das Modell wird daher auch als **Lotka-Volterrasches Differentialgleichungssystem** bezeichnet.

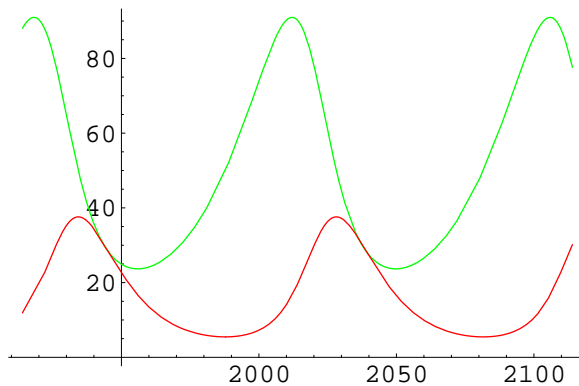
Da sich das System nicht explizit lösen lässt, müssen numerische Methoden zur Lösung verwendet werden. In Mathematica ist dies wieder mit dem Befehl `NDSolve` und vorgegebenen Werten möglich.

```
In[1]:= numerisch = NDSolve[{x'[t] == a x[t] - b x[t] y[t],
  y'[t] == -c y[t] + d x[t] y[t], x[1914] == 88.1, y[1914] == 11.9} /.
  {a -> 0.05, b -> 0.003, c -> 0.1, d -> 0.002}, {x[t], y[t]}, {t, 1914, 3000}];
```

Das hier verwendete Beispiel ist die Problematik, anhand derer die Modellierung eines Räuber-Beute-Systems erstmals vorgenommen wurde [KOE]: Volterra konnte allein durch die Daten des prozentualen Anteils der Haipopulation der gesamten Fische im Triester Seehafen zwischen den Jahren 1914 und 1923 die veränderten Populationsgrößen über die Jahre hinweg durch ein Modell beschreiben. Aus diesen Daten konnten dann die Werte $a = 0,05$, $b = 0,003$, $c = 0,1$ und $d = 0,002$ errechnet werden. Zu Beginn der Beobachtung, im Jahr 1914 betrug der Anteil der Haipopulation 11,9%, und der der übrigen Fischpopulationen 88,1%. Der gegebene Anfangswert ist also ($t_0 = 1914$; $x_0 = 88,1$, $y_0 = 11,9$).

```
In[2]:= Needs["Graphics`Colors`"]
```

```
In[3]:= P1 = Plot[Evaluate[{x[t], y[t]} /. numerisch],
  {t, 1914, 2114}, PlotStyle -> {Green, Red, Blue, Yellow}];
```



Die graphische Darstellung der interpolierten Lösungsfunktionen scheint eine Periodizität in der Wechselbeziehung der beiden Populationen von Räubern und Beute zu propagieren. Man kann aber sogar beweisen, dass die Periodizität für alle Lösungen des Voltterraschen Differentialgleichungssystems gilt und auch bei unterschiedlichen Anfangswerten erhalten bleibt.

Um obige Aussage zu zeigen, sollte eine allgemeine Gestalt für die Lösungen des Systems in Form einer Bahnkurve aufgestellt werden, d.h. der Parameter Zeit soll erst einmal aus dem System entfernt werden. Dabei geht man in folgenden Schritten vor: Unter Verwendung der Kettenregel und der Auflösung der Gleichung für x nach t , was formal einer Kehrwertbildung entspricht, ergibt sich:

$$\begin{aligned} \frac{dy}{dx} &\stackrel{KR}{=} \frac{dy}{dt} \frac{dt}{dx} \\ &= (-cy + dxy) \frac{1}{ax - bxy} \\ &= \frac{-cy + dxy}{ax - bxy} \\ &= \frac{y}{x} \left(\frac{-c + dx}{a - by} \right) \end{aligned}$$

Mit dem Kalül der Trennung der Variablen erhält man:

$$\frac{a - by}{y} dy = \frac{-c + dx}{x} dx .$$

Übergang zur Stammfunktion:

$$\int \frac{a - by}{y} dy = \int \frac{-c + dx}{x} dx .$$

Geschlossene Formen für die Stammfunktionen auf beiden Seiten:

$$a \ln(y) - by = -c \ln(x) + dx + C ,$$

mit einer Konstante C . Die Anwendung der Exponentialfunktion liefert

$$\begin{aligned} e^{a \ln(y) - by} &= e^{-c \ln(x) + dx + C} \\ \frac{y^a}{e^{by}} &= \frac{e^{dx}}{x^c} K \\ \frac{y^a}{e^{by}} \frac{x^c}{e^{dx}} &= K , \end{aligned}$$

dabei ist $K \in \mathbb{R}$ wieder eine Konstante. Diese Darstellung läßt sich aber natürlich nicht nach x und y auflösen.

Der Beweis für die Periodizität der Lösungen kann nun durch den Beweis des folgenden Lemmas geleistet werden:

Lemma 1.7 *Die Lösungen der Gleichung*

$$K = \frac{x^c}{e^{dx}} \frac{y^a}{e^{by}} \tag{1.24}$$

stellen geschlossene Kurven für $x, y > 0$ dar.

Beweis: Gemäß der Definition 1.3 ergeben sich für das Volterrasche System die Gleichgewichtspunkte aus den Nullstellen des Vektorfeldes

$$f(x, y) = \begin{pmatrix} ax - bxy \\ -cy + dxy \end{pmatrix} = \begin{pmatrix} x(a - by) \\ y(-c + dx) \end{pmatrix} .$$

In[1]:= `Solve[{a x - b x y == 0, -c y + d x y == 0}, {x, y}]`

Out[1]= `{{x -> 0, y -> 0}, {x -> c/d, y -> a/b}}`

Es gilt also $f(x, y) = 0$ für $x^* = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ und $x^{**} = \begin{pmatrix} c/d \\ a/b \end{pmatrix}$.

Sei nun nach [HEU]

$$K = \frac{x^c}{e^{dx}} \frac{y^a}{e^{by}} =: h(x)g(y) .$$

Durch Kurvendiskussion wird ersichtlich, dass die Graphen der Funktionen h und g sich qualitativ (bezüglich der Monotonie) etwa wie Glockenkurven verhalten, wobei das

einziges Maximum für $h(x) = \frac{x^c}{e^{dx}}$ bei $x = \frac{c}{d}$ und für $g(y) = \frac{y^a}{e^{by}}$ bei $y = \frac{a}{b}$ liegt.

Wir wollen nun im Folgenden die Werte der Funktionen h und g an ihren Maxima mit $h(\frac{c}{d}) =: M_h$ und $g(\frac{a}{b}) =: M_g$ bezeichnen. Um das Verhalten der Lösungen $x, y > 0$ der Gleichung (1.24) genauer beschreiben zu können, wird eine Fallunterscheidung durchgeführt:

1. Fall: $K > M_h M_g$

Dies ist nicht möglich nach der Definition von M_g als das Maximum von g . Es existiert keine Lösung von (1.24) für diesen Fall.

2. Fall: $K = M_h M_g$

Das ist der Fall, wenn h und g ihr Maximum annehmen. Also liefert dies die Lösung $x = \frac{c}{d}$, $y = \frac{a}{b}$. Die Lösung ist ein Gleichgewichtspunkt des Systems und beschreibt daher im Phasenportrait einen Punkt. Ein Punkt ist eine geschlossene Kurve.

3. Fall: $0 < K < M_h M_g$

Sei nun

$$h(x)g(y) =: \mu M_g \quad (1.25)$$

für ein μ mit $0 < \mu < M_h$. Für $h(x) = \mu$ existieren auf Grund des Monotonieverhaltens der Funktion genau zwei Lösungen x_1, x_2 mit $x_1 = \frac{c}{d}$ und $x_2 = \frac{a}{b}$.

Um Lösungen für Fall 3 zu finden, formen wir die Gleichung (1.25) um zu

$$g(y) = \frac{\mu}{h(x)} M_g .$$

Wir suchen also nun Lösungen für y in Abhängigkeit von x , indem eine weitere Fallunterscheidung für die verschiedenen qualitativen Möglichkeiten von x vorgenommen wird:

1. Fall: $x < x_1$ oder $x > x_2$

Dann gilt $h(x) < \mu$, somit $\frac{\mu}{h(x)} > 1$ und schließlich $g(y) > M_g$. Dies ist ein Widerspruch zur Definition von M_g .

2. Fall: $x = x_1$ oder $x = x_2$

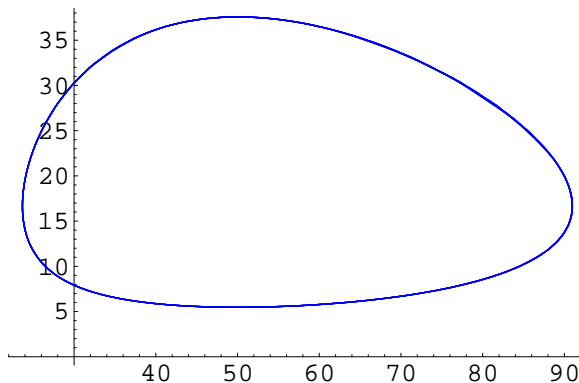
Es gilt dann $h(x) = \mu$. Daraus folgt $g(y) = M_g = g(\frac{a}{b})$ und man erhält eine Lösung $y = \frac{a}{b}$.

3. Fall: $x_1 < x < x_2$

Hieraus folgt $h(x) > \mu$ und damit $\frac{\mu}{h(x)} < 1$. Also gilt $g(y) < M_g$. Auf Grund des Monotonieverhaltens von g existieren also für jedes x aus dem Intervall (x_1, x_2) genau zwei Lösungen y_1, y_2 mit $0 < y_1 < \frac{a}{b} < y_2$. Für jedes $x \in (x_1, x_2)$ können somit je zwei Punkte $y_1(x), y_2(x)$ gefunden werden, die die Gleichung (1.24) erfüllen. Außerdem gilt $\lim_{x \rightarrow x_1} y_i(x) = \frac{a}{b}$ und $\lim_{x \rightarrow x_2} y_i(x) = \frac{a}{b}$ für $i = 1, 2$. Somit ist die Vereinigung der beiden Kurven (x, y_1) und (x, y_2) für $x \in [x_1, x_2]$ eine geschlossene Bahn. Für eine Lösung $x, y > 0$ des Lotka-Volterraschen Systems wurde also gezeigt, dass die im Phasendiagramm durch x und y beschriebene Punktmenge $(x, y(x))$ eine geschlossene Bahnkurve darstellt. \square

Die Darstellung der Lösungen des Lotka-Volterraschen Differentialgleichungssystems in einem Phasendiagramm als so genannte **Bahn** oder **Trajektorie** $(x(t), y(t))$ macht das periodische Verhalten durch die Geschlossenheit der Bahnkurve deutlich. Diese Bahnkurvendarstellung ist dabei zeitunabhängig. Die *Mathematica*-Funktion `ParametricPlot` zeichnet die durch $\{x(t), y(t)\}$ beschriebene Punktmenge.

```
In[3]:= ParametricPlot[Evaluate[{x[t], y[t]} /. numerisch], {t, 1914, 2600}, PlotStyle -> Blue]
```



```
Out[3]= - Graphics -
```

Definition 1.8 1. Eine Lösung $x(t; t_0, x_0)$ einer Differentialgleichung heißt **periodisch**, wenn ein $T > 0$ existiert, so dass gilt $x(t; t_0, x_0) = x(t + T; t_0, x_0)$. T bezeichnet dann die **Periode** der Differentialgleichung.

2. Sei $x(t; t_0, x_0)$ die periodische Lösung einer DGL, die die Größe einer Population beschreibt, und T die Periode der Lösung, so wird

$$\bar{x} := \frac{1}{T} \int_0^T x(t) dt$$

der **Mittelwert der Population** genannt.

Im folgenden Lemma soll nun noch gezeigt werden, dass für zwei Lösungen des Lotka-Volterra-Systems (1.22) und (1.23) die Durchschnittsgrößen der Populationen bereits durch den Gleichgewichtspunkt $\begin{pmatrix} \frac{c}{d} \\ \frac{a}{b} \end{pmatrix}$ gegeben sind.

Lemma 1.9 Seien $x(t), y(t)$ periodische Lösungen des Lotka-Volterraschen Differentialgleichungssystems mit Periode $T > 0$, so gilt $\bar{x} = \frac{c}{d}$ und $\bar{y} = \frac{a}{b}$.

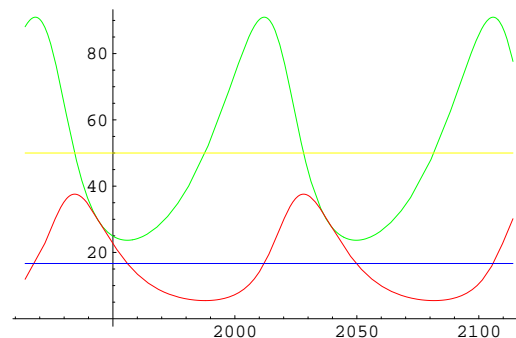
Beweis: Nach [KOE]. Wegen der Periodizität von y gilt $y(t) = y(t + T)$ für alle $t \in \mathbb{R}$.

Es folgt also

$$\begin{aligned}
 0 &= \frac{1}{T} \ln y(T) - \frac{1}{T} \ln y(0) \\
 &= \frac{1}{T} \ln y(t) \Big|_0^T \\
 &= \frac{1}{T} \int_0^T \frac{y'(t)}{y(t)} dt \\
 &= \frac{1}{T} \int_0^T \frac{-cy(t) + dx(t)y(t)}{y(t)} dt \\
 &= \frac{1}{T} \int_0^T (-c + dx(t)) dt \\
 &= -c + d\bar{x}.
 \end{aligned}$$

Also gilt $\bar{x} = \frac{c}{d}$. Für y läßt sich analog auf Grund der Periodizität von x , $\bar{y} = \frac{a}{b}$ zeigen. \square

Die durchschnittliche Populationsgröße ist also einer der Gleichgewichtspunkte. Dass dieses Gleichgewicht aber nie eintritt, zeigt die folgende Graphik, denn Räuber (rot) und Beute (grün) nehmen niemals gleichzeitig die Werte des Gleichgewichts (blau, gelb) an.



Auch die Geschlossenheit der Bahnkurve zeigt, dass das Verhältnis von Räuber zu Beute stets auf der Kurve bleibt und niemals sich auf den Gleichgewichtspunkt in der Mitte zubewegt. Man kann anschaulich nachvollziehen, dass das Gleichgewicht also nicht asymptotisch stabil ist.

Mit dem *Mathematica*-Befehl `ContourPlot` können die Lösungen des Lotka-Volterra-Systems auch als die Höhenlinien einer zweidimensionalen Funktion dargestellt werden. Jede Höhenlinie beschreibt dabei eine Lösung des Systems zu einem bestimmten Anfangswert.

```

In[1]:= numerisch = NDSolve[{x'[t] == a x[t] - b x[t] y[t],
    y'[t] == -c y[t] + d x[t] y[t], x[1914] == 88.1, y[1914] == 11.9} /.
    {a -> 0.05, b -> 0.003, c -> 0.1, d -> 0.002}, {x[t], y[t]}, {t, 1914, 3000}];

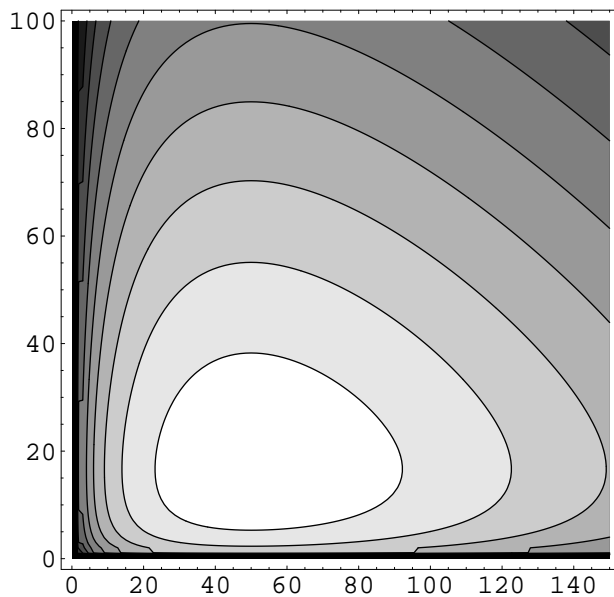
In[2]:= h[x_, y_] := ((y^a) (x^c)) / ((E^(b y)) (E^(d x)))

In[3]:= H = h[x, y] /. {a -> 0.05, b -> 0.003, c -> 0.1, d -> 0.002}

Out[3]= e-0.002x-0.003y x0.1 y0.05

In[4]:= ContourPlot[H, {x, 0, 150}, {y, 0, 100}, PlotPoints -> 100];

```



Das hier vorgestellte Lotka-Volterra Differentialgleichungssystem, also ein Räuber-Beute-System ohne jeglichen Einfluss von Dritten, ist allerdings relativ abstrakt und in der Realität äußerst selten vorzufinden. Um etwas realitätsnäher zu werden, müssen also weitere Einflussfaktoren zum Modell hinzugenommen werden.

Modell mit Berücksichtigung von externen, proportionalen Dezimierungseinflüssen

Unter der Annahme, dass sowohl auf die Räuber- als auch die Beutepopulation pro Zeiteinheit ein Reduktionsfaktor einwirkt, der für eine Konstante $e, f \in (0, 1)$ die Populationen um den Anteil $ex(t)$ bzw. $fy(t)$ reduziert, verändert sich das System von Lotka-Volterra zu

$$\begin{aligned}x'(t) &= ax(t) - bx(t)y(t) - ex(t), \\y'(t) &= -cy(t) + dx(t)y(t) - fy(t).\end{aligned}$$

Für dieses System ergeben sich natürlich die Populationsgleichgewichte $x^* = \left(\frac{c+e}{\frac{d}{a-e}} \right)$ und $x^{**} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$. Da sich in diesem Modell im Vergleich zum vorhergegangenen, al-

so nur die Parameter qualitativ etwas verändert haben, stellt ebenfalls $\bar{x} = \frac{c+e}{d}$ und $\bar{y} = \frac{a-e}{b}$ den Mittelwert der Populationen dar. Ein regelmäßiger reduzierender Eingriff in Räuber- und Beutepopulation erweist sich also durchschnittlich als eine Vergrößerung der Beute- und eine Verringerung der Räuberpopulation.

Dieses Prinzip, das auch *Volterrasches Prinzip* genannt wird, hat Auswirkungen in vielen biologischen Systemen. Werden zum Beispiel bei Ungezieferbehandlungen Pestizide verwendet, welche sowohl die Beuteart, als auch die Räuber negativ beeinträchtigen, so kann dies den gegenteiligen Effekt des ursprünglichen Ziels bewirken, da sich das Ungeziefer durch die Behandlung sogar vermehrt.

Modell mit Berücksichtigung von innerartlicher Konkurrenz

Das Räuber-Beute-Modell, welches in beiden Populationen von begrenztem Lebensraum und damit Konkurrenz zwischen den Individuen einer Art ausgeht, wird aufgestellt durch das Differentialgleichungssystem

$$\begin{aligned} x'(t) &= ax(t) - bx(t)y(t) - ex(t)^2, \\ y'(t) &= -cy(t) + dx(t)y(t) - fy(t)^2, \end{aligned}$$

mit den zusätzlichen Konkurrenzkonstanten $e, f > 0$. Für dieses System ergeben sich mit *Mathematica* die Gleichgewichte:

In[1]:= `Solve[{a x - b x y - e x^2 == 0, -c y + d x y - f y^2 == 0}, {x, y}]`

Out[1]= `{ {x -> 0, y -> 0}, {x -> a/e, y -> 0}, {x -> -b c - a f / (b d + e f), y -> -a d + c e / (b d + e f)}, {y -> -c/f, x -> 0} }`

Es kommt also noch ein dritter positiver, potenziell relevanter Gleichgewichtspunkt hinzu, falls $ad > ce$, denn negative Gleichgewichtspunkte sind verständlicherweise nicht von Interesse. Die Bedingung $ad > ce$ ist also gegeben, wenn die Wachstumsrate der Beute und der Vorteil der Räuber am Vorhandensein der Beute groß genug ist. Die allgemeine Stabilitätsdiskussion der Eigenwerte scheint sehr aufwändig.

In[1]:= `Ja = (a - b y - 2 e x - b x
 d y - c + d x - 2 f y);
EWE = Eigenvalues[Ja];
EWE = EWE /. {x -> -b c - a f / (b d + e f), y -> -a d + c e / (b d + e f)}`

Out[3]= `{ 1/2 (a - c + b (-a d + c e) / (b d + e f) + 2 (-a d + c e) f / (b d + e f) - d (-b c - a f) / (b d + e f) + 2 e (-b c - a f) / (b d + e f) -
 sqrt((-a + c - b (-a d + c e) / (b d + e f) - 2 (-a d + c e) f / (b d + e f) + d (-b c - a f) / (b d + e f) - 2 e (-b c - a f) / (b d + e f))^2 -
 4 (-a c + 2 b (-a d + c e)^2 f / (b d + e f)^2 + 4 e (-a d + c e) f (-b c - a f) / (b d + e f)^2 - 2 d e (-b c - a f)^2 / (b d + e f)^2 -
 b c (-a d + c e) / (b d + e f) + 2 a (-a d + c e) f / (b d + e f) - a d (-b c - a f) / (b d + e f) - 2 c e (-b c - a f) / (b d + e f)))) ,
 1/2 (a - c + b (-a d + c e) / (b d + e f) + 2 (-a d + c e) f / (b d + e f) - d (-b c - a f) / (b d + e f) + 2 e (-b c - a f) / (b d + e f) +
 sqrt((-a + c - b (-a d + c e) / (b d + e f) - 2 (-a d + c e) f / (b d + e f) + d (-b c - a f) / (b d + e f) - 2 e (-b c - a f) / (b d + e f))^2 -
 4 (-a c + 2 b (-a d + c e)^2 f / (b d + e f)^2 + 4 e (-a d + c e) f (-b c - a f) / (b d + e f)^2 - 2 d e (-b c - a f)^2 / (b d + e f)^2 -
 b c (-a d + c e) / (b d + e f) + 2 a (-a d + c e) f / (b d + e f) - a d (-b c - a f) / (b d + e f) - 2 c e (-b c - a f) / (b d + e f)))) }`

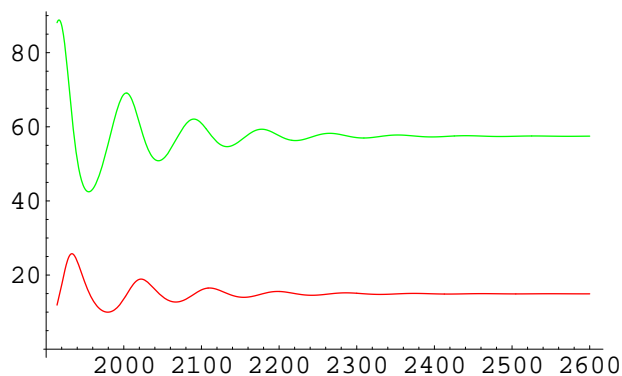
Deswegen beschränken wir uns hier auf die Betrachtung eines konkreten Beispiels.

Verändert man das Beispiel des Triester Seehafens also so, dass die Bedingung $ad > ce$

erfüllt wird, so ist der dritte Gleichgewichtspunkt $x^{***} = \begin{pmatrix} \frac{-bc-af}{bd+ef} \\ \frac{-ad+ce}{bd+ef} \end{pmatrix}$ sogar asymptotisch stabil. Sollen die Werte a, d, c dabei fest gelasse werde, so muss der Konkurrenzparameter $e = 0,00009$ der Räuber allerdings sehr klein werden. Der Konkurrenzparameter für die Beute spielt bei dieser Bedingung keine Rolle, er wird einfach $f = 0,001$ gesetzt. Nach dem obligatorischen Laden des Packages `Graphics`Colors`` berechnet *Mathematica* das Beispiel.

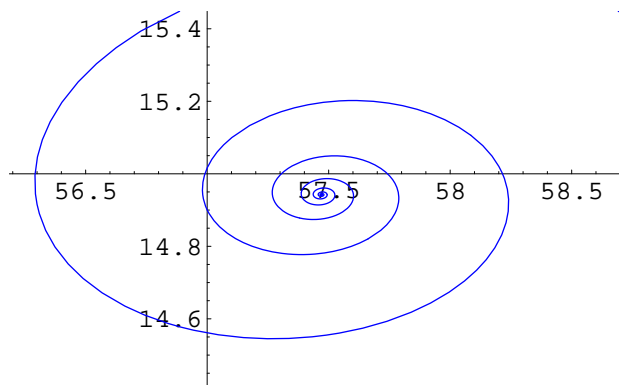
```
In[2]:= numerisch = NDSolve[
  {x'[t] == a x[t] - b x[t] y[t] - e x[t]^2, y'[t] == -c y[t] + d x[t] y[t] - f y[t]^2,
   x[1914] == 88.1, y[1914] == 11.9} /. {a -> 0.05, b -> 0.003, c -> 0.1,
   d -> 0.002, e -> 0.00009, f -> 0.001}, {x[t], y[t]}, {t, 1914, 3000}];
```

```
In[3]:= Plot[Evaluate[{x[t], y[t]} /. numerisch],
  {t, 1914, 2600}, PlotStyle -> {Green, Red, Blue, Yellow}]
```



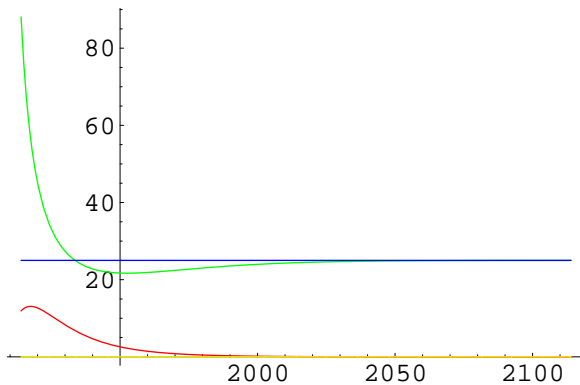
Für dieses Beispiel ist es auch interessant sich die Lösungen für x und y , dargestellt im Phasenportrait anzusehen. Hier zeigt sich deutlich die Attraktivität des stabilen Gleichgewichts, indem sich die Bahnkurve in Form einer Spirale dem Gleichgewicht immer weiter annähert.

```
In[4]:= ParametricPlot[Evaluate[{x[t], y[t]} /. numerisch],
  {t, 1914, 3000}, PlotStyle -> {Blue, Red}]
```



Wählt man allerdings die beiden Konkurrenzparameter in derselben Größenordnung, so existieren als positive Gleichgewichte nur die beiden Punkte $x^* = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ und $x^* = \begin{pmatrix} \frac{a}{e} \\ 0 \end{pmatrix}$. Dies bedeutet, dass in diesem Modell die Räuber langfristig aussterben werden.

Mit den gleichen Daten des vorherigen Beispiels unter Hinzunahme der beiden Konkurrenzparameter $e = 0.002$ und $f = 0.001$ verhält sich das Populationswachstum der Räuber (rote Linie) und der Beute (grüne Linie) auch gemäß dieser Vorhersage und findet sich auf lange Sicht in dem stabilen Gleichgewicht $x^* = \begin{pmatrix} \frac{a}{e} \\ 0 \end{pmatrix}$ des Systems ein.



1.2.2 Wachstumsmodelle konkurrierender Arten

Die nächste Art der Wechselbeziehung zwischen zwei Arten, welche hier behandelt werden soll, ist die der Konkurrenz um gemeinsam genutzte Ressourcen. Ausgangslage soll dabei sein, dass beide Arten ein logistisches Wachstum in der Abwesenheit der jeweiligen anderen Art aufweisen. Die beiden Arten beeinflussen gegenseitig ihr Wachstum negativ. Dabei kann die Beeinträchtigung durch die konkurrierende Nachbarart jeweils von verschiedenem Ausmaß sein. Das entsprechende Differentialgleichungssystem wird in ähnlicher Weise wie das Lotka-Volterra-Modell aufgestellt. Der entscheidende Term der Konkurrenz wird in der DGL durch eine Subtraktion eines Anteil der Konkurrenzpopulation angegeben:

$$x'(t) = ax(t) - bx(t)y(t) - ey(t), \tag{1.26}$$

$$y'(t) = cy(t) - dx(t)y(t) - fx(t). \tag{1.27}$$

Die Gleichgewichtspunkte des Systems sind durch

$$\begin{pmatrix} x^+ \\ y^+ \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} x^* \\ y^* \end{pmatrix} = \begin{pmatrix} \frac{ac-ef}{ad+bf} \\ \frac{ac-ef}{bc+de} \end{pmatrix}$$

gemäß der unten stehenden Rechnung gegeben.

```

In[1]:= Needs["Graphics`Colors`"]

In[2]:= S = Solve[{a x - b x y - e y == 0, c y - d x y - f x == 0}, {x, y}]

Out[2]= {{x -> 0, y -> 0}, {x ->  $\frac{a c - e f}{a d + b f}$ , y ->  $\frac{a c - e f}{b c + d e}$ }}

In[3]:= v =  $\frac{a c - e f}{a d + b f}$  /. {a -> 0.05, b -> 0.003, c -> 0.1, d -> 0.002, e -> 0.002, f -> 0.001};

In[4]:= w =  $\frac{a c - e f}{b c + d e}$  /. {a -> 0.05, b -> 0.003, c -> 0.1, d -> 0.002, e -> 0.002, f -> 0.001};

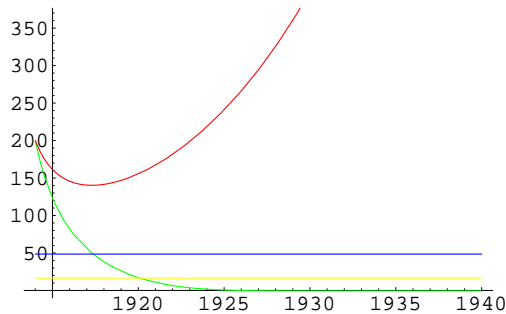
In[5]:= numerisch =
  NDSolve[{x'[t] == a x[t] - b x[t] y[t] - e y[t], y'[t] == c y[t] - d x[t] y[t] - f x[t],
    x[1914] == 200, y[1914] == 200} /. {a -> 0.05, b -> 0.003, c -> 0.1,
    d -> 0.002, e -> 0.002, f -> 0.001}, {x[t], y[t]}, {t, 1914, 2000}];

In[6]:= P1 = Plot[Evaluate[{x[t], y[t]} /. numerisch],
  {t, 1914, 1940}, PlotStyle -> {Green, Red, Blue, Yellow}];

In[7]:= P2 = Plot[{v, w}, {t, 1914, 1940}, PlotStyle -> {Blue, Yellow}];

In[8]:= Show[P1, P2]

```



Der nichttriviale Gleichgewichtspunkt ist bei diesem Beispiel instabil. Man sieht, der bevorzugte Konkurrent ($a < c$, $f < e$) verdrängt seinen Rivalen bei gleicher Anfangsgröße in kurzer Zeit vollständig und geht danach zu exponentiellem Wachstum über.

1.2.3 Wachstumsmodelle für Arten in Symbiose

Noch unbehandelt geblieben ist schließlich eine positive Wechselbeziehung zwischen verschiedenen Arten, eine so genannte Symbiose oder Mutualismus zwischen zwei Arten. Dies bedeutet, dass beide Arten von einem Zusammenleben im selben Lebensraum profitieren. In einem Differentialgleichungssystem mit diesen Grundlagen manifestiert sich diese Beziehung natürlich durch positive Terme. Man geht davon aus, dass der Vorteil einer Art umso größer ist, desto mehr symbiotische Partner vorhanden sind. Selbstverständlich kann dieser Vorteil für jede Seite der Symbiose dabei unterschiedlich stark sein.

$$x'(t) = ax(t) + ex(t)y(t), \quad (1.28)$$

$$y'(t) = cy(t) + fx(t)y(t). \quad (1.29)$$

Da in diesem einfachen Modell nur positive Terme auftreten, hat dies für das Wachstumsverhalten der beteiligten Populationen ein unbeschränktes, sich sogar sehr schnell

beschleunigendes Wachstum zur Folge. Für die Modellierung einer realen Situation müssen daher noch begrenzende Faktoren in das Modell mitaufgenommen werden.

System mit Mutualismus und Begrenzungsfaktoren

Für realistischere Varianten einer Modellierung sollen also wieder Dezimierungen der Population durch eine Begrenzung der Ressourcen und des Lebensraums miteinbezogen werden. Man nimmt wieder an, dass beide Arten ohne ihren Symbiosepartner ein logistisches Wachstumsverhalten zeigen. Das Modell (1.28), (1.29) verändert sich damit zu:

$$x'(t) = ax(t) - bx(t)^2 + ex(t)y(t) \quad (1.30)$$

$$y'(t) = cy(t) - dy(t)^2 + fx(t)y(t) \quad (1.31)$$

Für dieses System gibt es auf Grund seiner Struktur, da die rechten Seiten beide Polynome in x bzw. y vom Grad 3 sind, sogar vier Gleichgewichtspunkte:

```
In[1]:= Solve[{0 == a x + e x y - b x^2, 0 == c y + f x y - d y^2}, {x, y}]
```

```
Out[1]= {{x -> 0, y -> 0}, {x -> a/b, y -> 0}, {x -> -a d + c e / -b d + e f, y -> -b c - a f / b d - e f}, {y -> c/d, x -> 0}}
```

Neben dem üblichen Nullpunkt ergeben sich hier auch zwei Gleichgewichte, welche das vollständige Aussterben jeweils einer der beiden Arten propagieren. So bleibt letztlich nur ein Equilibrium zur weiteren Untersuchung übrig, welches ein reales Gleichgewicht des Systems bei zwei überlebenden Arten beschreibt.

Für das Gleichgewicht

$$\begin{pmatrix} x^* \\ y^* \end{pmatrix} = \begin{pmatrix} -\frac{ad-ce}{bd-ef} \\ -\frac{bc-af}{bd-ef} \end{pmatrix}$$

muss, um die Positivität sicher zu stellen lediglich $bd > ef$ gefordert werden. Für unser Beispiel mit den üblichen Parametern muss diese Bedingung bei der Festlegung der Symbioseparameter e und f also berücksichtigt werden.

```
In[2]:= Needs["Graphics`Colors`"]
```

```
In[3]:= Clear[numerisch, reg]
```

```
reg = {a -> 0.05, b -> 0.003, c -> 0.1, d -> 0.002, e -> 0.0008, f -> 0.001};
```

```
numerisch = NDSolve[
```

```
{x'[t] == a x[t] - b x[t]^2 + e x[t] y[t], y'[t] == c y[t] - d y[t]^2 + f x[t] y[t],
```

```
x[1914] == 30, y[1914] == 30} /. reg, {x[t], y[t]}, {t, 1914, 3000}];
```

```
In[6]:= bd /. reg
```

```
ef /. reg
```

```
Out[6]= 6. × 10-6
```

```
Out[7]= 8. × 10-7
```

Eine Wahl von $e = 0,0008$ und $f = 0,001$ erfüllt also die Bedingung und man kann mit den folgenden Rechnungen zeigen, dass das hier untersuchte Gleichgewicht asym-

ptotisch stabil ist.

```
In[8]:= - $\frac{ad+ce}{bd+ef}$  /. reg
        - $\frac{bc-af}{bd-ef}$  /. reg
```

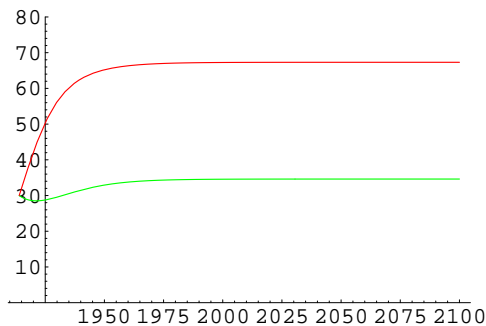
```
Out[8]= 34.6154
```

```
Out[9]= 67.3077
```

```
In[10]:= J =  $\begin{pmatrix} a-2bx+ey & ex \\ fy & c-2dy+fx \end{pmatrix}$ ;
        Z = J /. {x -> - $\frac{ad+ce}{bd+ef}$ , y -> - $\frac{bc-af}{bd-ef}$ };
        Eigenvalues[Z] /. reg
```

```
Out[12]= {-0.165063, -0.0733986}
```

```
In[13]:= Plot[Evaluate[{x[t], y[t]} /. numerisch],
             {t, 1914, 2100}, PlotStyle -> {Green, Red}, PlotRange -> {0, 80}];
```



Um also ein attraktiv und stabiles Gleichgewicht für zwei Populationen in einer mutualistischen Beziehung zu erhalten, muss das Produkt der begrenzenden Faktoren b und d der beiden Arten das Produkt der Parameter des Vorteils auf Grund der Symbiose e und f übertreffen.

1.3 Wachstumsmodelle für mehrere Arten in Wechselbeziehungen

In den vorangegangenen Abschnitten des Kapitels wurden also verschiedene Grundmodelle des Wachstums einer Art und schließlich Wachstumsmodelle von zwei Arten in unterschiedlichen Interaktionsweisen vorgestellt. In der freien Wildbahn jedoch lässt sich nur in seltenen Fällen ein Szenario finden, in dem diese Grundmodelle für eine Simulation der Verhältnisse ausreichen. Dagegen sollte für eine realitätsnahe Modellierung eine Kombination der Grundmodelle verwendet und ein System aufgestellt werden, welches alle beteiligten Parteien berücksichtigt und ihre Verhältnisse untereinander darstellt.

Allgemein wird ein Differentialgleichungssystem für ein System von n interagierenden

Arten in einem gemeinsamen Lebensraum in der folgenden Form formuliert:

$$\frac{dx_i}{dt} = a_i x_i + \sum_{j=1}^n b_{ij} x_i x_j, \quad i = 1, \dots, n \quad (1.32)$$

Dabei beschreiben die Parameter a_i die Wachstumsrate der Art x_i für sich betrachtet, und die $b_{ii} < 0$ stellen den Begrenzungsfaktor durch innerartliche und umweltbedingte Beschränkungen dar. Die Koeffizienten b_{ij} mit $i \neq j$ charakterisieren durch ihr Vorzeichen und ihre Größe die Wirkungsweise der j ten auf die i te Population.

Um eine Population unter möglichst realen Bedingungen zu untersuchen, müssen dabei natürlich auch noch ganz andere Aspekte, wie die Altersstruktur, die räumliche Ausbreitung einer Art und temporäre Bedingungen, wie etwa der Einfluss von Jahreszeiten in die Betrachtungen miteinfließen.

Natürlich kann ein Modell, welches alle beteiligten Wechselwirkungen in biologischen Systemen berücksichtigt sicherlich leichter den Anspruch auf Wahrheitstreue erheben. Allerdings wird andererseits in einem solchen System die Verflechtung der beteiligten Größen auch immer stärker und damit die Bestimmung von angemessenen Parametern mit einem mathematischen Modell sehr erschwert.

Verwendete Literatur

In diesem Kapitel habe ich mich bei der Einführung der verschiedenen Populations-Modelle hauptsächlich auf die Quellen [KOE] und [MUR] gestützt. Für die theoretische Untermauerung zu gewöhnlichen DGLen und zur Stabilitätstheorie wurden [AUL], [HEU], [BRA] und [GRU] und für die Parameterschätzung [YSH] und [GEO] herangezogen.

Kapitel 2

Neurophysiologie

2.1 Einführung in die Thematik

Nervenzellen sind Zellen, welche für die Übertragen von elektrischer Information im Körper von Lebewesen spezialisiert sind. Sie können Kodierungen von Sinneseindrücken der Außenwelt aufnehmen und an höhere Verarbeitungsinstanzen weiterleiten, Handlungsanweisungen des Gehirns an den Körper übermitteln oder dem Informationsaustausch zwischen Organen dienen.

Die Informationsübertragung geschieht in Form von elektrischen Signalen, welche pulsartig von einer Nervenzelle an andere Zellen weitergegeben werden. Durch Intensität, Dauer und Häufigkeit der Impulse wird die übertragene Information präzisiert.

Der Aufbau eines Neurons gliedert sich, wie in der unten stehenden Abbildung 2.1 in die folgenden Hauptbestandteile, den Zellkörper mit einem Zellkern und den Vorrichtungen der Verbindung zu anderen Zellen, dem Axon und den Dendriten [YSH]. Einerseits werden Informationen angrenzender Zellen von den Dendriten aufgenommen, andererseits wird die im Zellkern 'verarbeitete' Information zunächst über das Axon und an dessen Enden über so genannte Synapsen an die Dendriten oder direkt den Zellkörper anderer Zellen weitergeleitet.

Für die Entstehung und Weiterleitung von elektrischen Signalen in Neuronen ist die

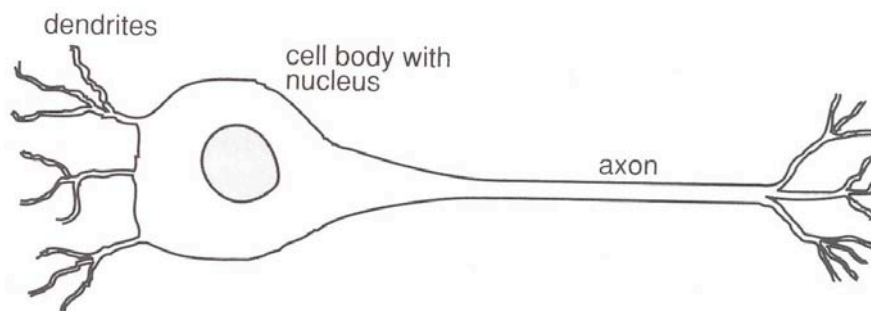


Abbildung 2.1: Der Aufbau eines Neurons

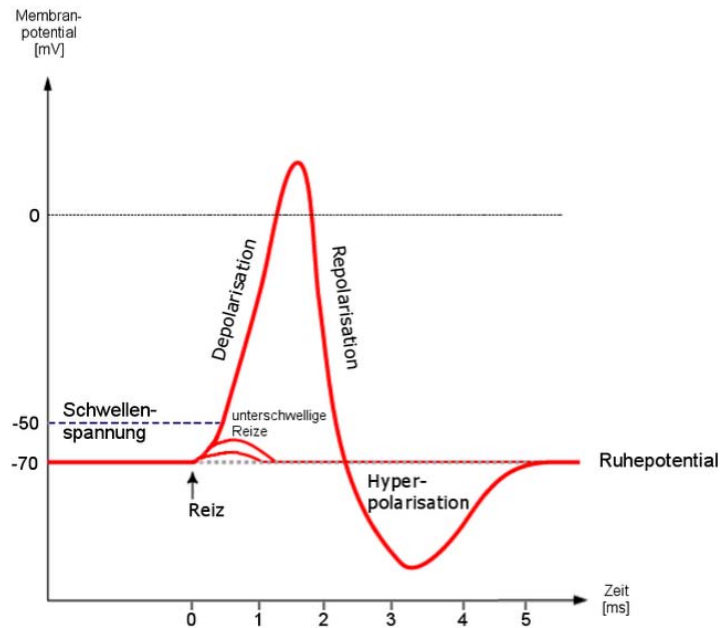


Abbildung 2.2: Der Verlauf eines Aktionspotentials

Membran der Zelle verantwortlich. Auf Grund der unterschiedlichen Permeabilität der Membran für verschiedene Ionen innerhalb und außerhalb der Zelle ist ein Neuron im Ruhezustand, d.h. ohne äußere Erregung leicht negativ polarisiert. Dieses Ladungsgefälle zwischen dem Intra- und Extrazellularraum wird in Millivolt angegeben, da gerade Volt die elektrische Spannung mißt, welche zwischen dem negativen Inneren und dem positiven Äußeren der Zelle herrscht. Diese Spannung wird als **Membranpotential** bezeichnet. Im Ruhezustand beträgt dieses Potential etwa -70 mV .

Die Informationsübertragung zwischen den Neuronen geschieht über die Bildung von so genannten **Aktionspotentialen**. Durch das Andocken von Neurotransmittern an den Dendriten, welche von Synapsen benachbarter, erregter Zellen ausgeschüttet werden, werden die Kanäle für Na-Ionen in der Membran kurzzeitig geöffnet, so dass es zu einer Depolarisation kommt. Das Potential der Zelle nähert sich dem Ausgleich. Ist eine gewisse Schwelle erreicht, läuft eine 'Alles-oder-Nichts'-Reaktion ab. Die Ionen-Kanäle öffnen sich für die verschiedenen beteiligten Ionen in einer bestimmten Reihenfolge, so dass das Membranpotential so weit depolarisiert wird bis ein Pik des Potentials bei etwa $+60\text{ mV}$ erreicht wird. Danach fällt das Potential der Membran in der Repolarisierungsphase wieder ab, schießt dabei aber über die Ausgangslage des Ruhepotentials hinaus und gelangt erst nach einer kurzen Hyperpolarisation des Potentials wieder zum Ruhepotential zurück. Dann erst ist die Nervenzelle wieder von neuem erregbar. Abbildung 2.2 zeigt den Verlauf eines Aktionspotentials [WWW].

2.2 Das Modell: VCON - Voltage Controlled Oscillator Neuron

Das Neuronen-Modell VCON, welches im Folgenden vorgestellt werden soll, wurde von F. C. Hoppensteadt entwickelt. Im Unterschied zu den vielen existierenden Modellen für Neuronen betont dieses Modell besonders den Aspekt von Neuronen als biologische Uhren, welche unter entsprechenden Einflüssen ihre Phase, und damit ihren Rhythmus verändern. Das Ziel des Modells ist es, die oszillierenden Potentialveränderungen eines Aktionspotentiale feuernenden Neurons zu modellieren, welche durch De- und Repolarisierungen der Zellmembran, ausgelöst wiederum durch Aktionspotentiale oder unterschwellige Reize von Nachbarneuronen, kontrolliert werden.

Physikalische Grundlagen

Elektrische Schaltkreise spielen für die Erstellung des Modells eine grundlegende Rolle. Die einzelnen physikalischen Grundprinzipien, welche als Komponenten in das Modell eingehen, werden nun vorgestellt.

Definition 2.1 *Ein VCO (voltage controlled oscillator), ein spannungsgesteuerter Oszillator ist ein Modell, welches durch die Eingabe eines Spannungsparameters V_{in} in der Einheit Volt die Frequenz seiner Ausgabe V , ebenfalls in Volt, in einem gewissen Verhältnis verändert.*

Für das Neuronen-Modell hier wird ein bestimmter VCO mit einigen festen Annahmen verwendet. Dabei ist $V : \mathbb{R} \rightarrow \mathbb{R}$ eine differenzierbare und periodische Funktion, deren Argumentfunktion die Phase $x : \mathbb{R} \rightarrow \mathbb{R}$ ist.

$$V_{in} \rightarrow [VCO] \rightarrow V(x(t))$$

Die Phase $x(t)$ ist die unbekannte Größe. Gilt $V_{in} \in V(x(t))$, d.h. liegt V_{in} im Bild von $V \circ x$, so steht die Phase im Verhältnis zu der Spannungseingabe V_{in} gemäß der Differentialgleichung

$$\frac{dx}{dt} = \omega + \sigma V_{in}. \quad (2.1)$$

Dabei steht ω für die *zentrale Frequenz* und σ ist der Faktor, welcher die *Sensitivität des VCOs* beschreibt. Die zentrale Frequenz wird hier für ein VCO als konstant angenommen, desweiteren wird auch für die Sensitivität die Annahme $\sigma = 1$ getroffen. Die Differentialgleichung (2.1) kann durch einfaches Integrieren beider Seiten gelöst werden:

$$x(t) = x(0) + \omega t + \int_0^t V_{in}(s) ds \quad (2.2)$$

Ist ω oder V_{in} besonders groß, so bewirkt dies ein besonders starkes Oszillieren der Ausgabe des VCO. Entspricht andererseits die Eingabespannung bereits der zentralen Frequenz, d.h. V_{in} ist konstant und es gilt $V_{in} = -\omega$, so ist auch die Phase x konstant.

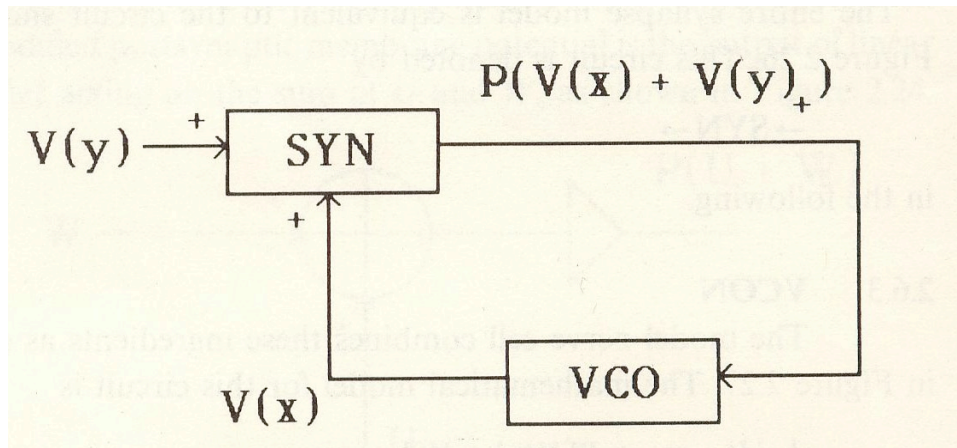


Abbildung 2.3: Eine VCON-Rückkopplungsschleife

Betrachtet man nun die Situation, dass die Eingabe des VCOs nicht nur aus einer beliebigen Spannung besteht, sondern auch die Ausgabe des VCOs im nächsten Input miteinbezogen wird, so erhält man eine **VCO-Rückkopplungsschleife**. Diese Schleife ist erster Ordnung, wenn der Output ungefiltert zu V_{in} addiert wird und diese Summe nun die Eingabe darstellt. Die Differentialgleichung, welche die Phase x beschreibt, bekommt dadurch die folgende Form:

$$\frac{dx}{dt} = \omega + V_{in} + V(x(t)). \quad (2.3)$$

Definition 2.2 Ein **Low-pass Filter** ist eine Funktion, welche nur Frequenzen bis zu einer festgelegten cut-off-frequency passieren läßt. Höhere Frequenzen als diese werden nicht übertragen. Nach dem Ohm'schen Gesetz und der charakteristischen Eigenschaft eines Kondensators wird ein Low-pass Filter durch die Gleichung

$$RC \left(\frac{dV_{out}}{dt} \right) + V_{out} = V_{in} \quad (2.4)$$

beschrieben.

2.2.1 Ein einzelnes Neuron

Mit dem VCON-Modell soll die Situation eines Neurons unter dem Einfluß einer Synapse modelliert werden. Das VCON setzt sich dabei aus zwei Komponenten zusammen. Durch eine VCO-Rückkopplungsschleife wird einerseits das Membranpotential des Neurons dargestellt, andererseits wird die Erregungsübertragung über die einwirkende Synapse, in Form von Neurotransmittern, welche durch den synaptischen Spalt diffundieren und an der Zellmembran Potentialveränderungen bewirken, durch einen Low-pass Filter

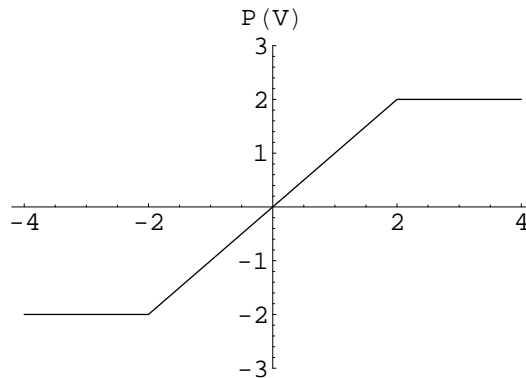


Abbildung 2.4: Ein linearer Amplifier

modelliert. Durch die Verbindung dieser beiden Aspekte entsteht das VCON-Modell. Das Grundschema eines VCONs ist in Abbildung 2.3 mit den beiden Komponenten VCO und SYN dargestellt [HOP].

Das Modell für den Zellkörper, eine VCO-Rückkopplungsschleife wurde bereits im vorangegangenen Abschnitt demonstriert. Als Ausgabefunktion des VCOs wird nun von einer 2π -periodischen, stetig differenzierbaren Funktion ausgegangen.

An Hand eines Low-pass Filters soll nun die synaptische Übertragung modelliert werden. Bei dieser Übermittlung von Aktionspotentialen von einer Zelle zur nächsten über die Neurotransmitter spielen deren chemikalische Prozesse im synaptischen Spalt eine gewisse Rolle. Die Konzentration der ausgeschütteten Neurotransmitter im Spalt zwischen Synapse und Dendrit dezimiert sich mit der Zeit durch Diffusion der Moleküle in die Peripherie k_{dif} und durch das Andocken an der postsynaptischen Membran der Zelle k_{post} . Das ankommende Potential W , welches von der erregungsübertragenden Zelle über die Synapse an die Nachbarzelle übermittelt werden soll, ändert sich also auf Grund der Übertragung gemäß der Differentialgleichung eines Low-pass Filters

$$RC \frac{dW}{dt} = -W + W_{in}. \quad (2.5)$$

W_{in} ist hierbei das Potential des übermittelnden Neurons und $RC = k_{dif} + k_{post}$. Für die Ausschüttung der Neurotransmitter ist jedoch nur der positive Anteil von W_{in} von Interesse. Deshalb wird für W_{in} eine Diode, i.e. eine Funktion d mit $d(W_{in}) = \max(W_{in}, 0)$, die nur den positiven Teil von W_{in} oder Null ausgibt, vorgeschaltet.

Zur Verbindung der beiden Modellkomponenten zum VCON-Modell wird nun noch ein linearer Amplifier eingefügt. Dieser bewirkt, bevor der Output des Low-pass Filters mit dem ursprünglichen Potential der Zelle aufaddiert in die VCO-Rückkopplungsschleife eingeht, dass deren Summe in niedrigeren Wertebereichen identisch übertragen wird und nur extreme Potentialwerte, um den Rahmen der physiologischen Grenzen nicht zu überschreiten, durch einen Maximalwert genormt werden. Abbildung 2.4 (Mathematica) zeigt einen charakteristischen linearen Amplifier mit beliebigen Werten. In vielen numerischen Simulationen erwies sich der Tangenshyperbolicus als Amplifier mit ähnlichen Eigenschaften für diese Aufgabe geeignet und wird deshalb im Folgenden verwendet.

Das Potential der erregungsübertragenden Zelle muß natürlich nicht konstant sein, sondern kann ebenfalls von einer Phase $y : \mathbb{R} \rightarrow \mathbb{R}$ abhängen und somit selbst bereits ein oszillierendes Verhalten aufweisen.

Man erhält schließlich die beiden Gleichungen, welche das VCON-Neuronen-Modell darstellen:

$$\frac{dx}{dt} = \omega + P(V(x) + V_{in}) \quad (2.6)$$

$$RC \frac{dW}{dt} = -W + d(W_{in}(y)). \quad (2.7)$$

Die Miteinbeziehung der chemikalischen Aspekte in die Modellierung des Neurons bewirkt allerdings eine starke Verkomplizierung des Modells, deshalb werden im Folgenden mit der Annahme $k_{dif} + k_{post} = 0$ die molekularen Prozesse im synaptischen Spalt außer Acht gelassen. Die Gleichung (2.5) vereinfacht sich somit zu $W = d(W_{in})$, und unter dieser Voraussetzung lassen sich auch die beiden Gleichungen (2.6) und (2.7) des VCON zu einer einzigen Gleichung zusammenfassen, wenn man davon ausgeht, dass die modellierte Situation zwei gleichartige Neuronen betrachtet und damit $W = V$ gilt, d.h. die beiden Potentialfunktionen der Zellen stimmen überein:

$$\frac{dx}{dt} = \omega + P(V(x) + V(y)). \quad (2.8)$$

Für die Potentialfunktion $V : \mathbb{R} \rightarrow \mathbb{R}$ soll insbesondere festgelegt werden, $V(x) = A \cos(x)$ für ein $A \in \mathbb{R}$.

Mit *Mathematica* kann dann später diese Differentialgleichung mit dem Befehl `NDSolve` nach einer Wahl des Stimulus und eines Anfangswerts einfach gelöst und visualisiert werden.

Interessant für die Untersuchung des Modells sind nun die unterschiedlichen Arten von Stimulationen des modellierten Neurons. Separat betrachtet werden sollen im Folgenden die drei Fälle:

1. Die Zelle im Ruhezustand, ohne äußere Erregung
2. Ein einmaliger Stimulationsimpuls der Zelle
3. Oszillierende Erregungen des Neurons

1. Das Neuron im Ruhezustand

Bleibt die Zelle unbeeinflusst von jeglichen Stimuli, so pendelt sich ihr Potential im Ruhepotential von etwa -70 mV ein. Gleichung (2.8) des VCON Modells bekommt also die Gestalt

$$\frac{dx}{dt} = \omega + P(V(x)). \quad (2.9)$$

Das Modell soll das Ruhepotential der Zelle modellieren, d.h. es soll eine Phase x gefunden werden, für die das Potential des Neurons konstant das Ruhepotential annimmt. Das Potential $V(x) = A \cos(x)$ ist aber genau dann konstant, wenn die Phase x konstant ist, und dies gilt genau dann, wenn die rechte Seite von (2.9) nach den Einsetzungen von V und P Null gesetzt wird. An dieser Stelle wird die Forderung $\omega \in P(V(x)) = \tanh(A \cos(x))$ nötig.

$$0 = \omega + \tanh(A \cos(x)) \quad (2.10)$$

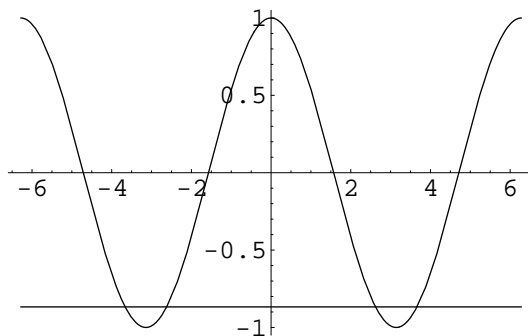
$$\tanh(A \cos(x)) = -\omega \quad (2.11)$$

$$A \cos(x) = \operatorname{arctanh}(-\omega) \quad (2.12)$$

$$\cos(x) = -\frac{\operatorname{arctanh}(\omega)}{A} \quad (2.13)$$

Unter der Bedingung $1 \leq \left| \frac{\operatorname{arctanh}(\omega)}{A} \right|$ gibt es im Intervall 2π auf Grund der Periodizität des Cosinus stets zwei Lösungen in (2.13) für x .

```
In[1]:= A = 1;
        ω = .7;
        Plot[{Cos[x], -ArcTanh[ω] / A}, {x, -2 π, 2 π}]
```



```
Out[3]= - Graphics -
```

Es gibt also zwei Schnittpunkte x^* und x^{**} in $[0, 2\pi]$ und für ihr Verhältnis untereinander gilt $x^{**} = 2\pi - x^*$.

```
In[4]:= S = Solve[ω + Tanh[A Cos[x]] == 0, x]
```

```
Solve::ifun : Inverse functions are being used by Solve, so some
solutions may not be found; use Reduce for complete solution information. More...
```

```
Out[4]= {{x -> -2.62055}, {x -> 2.62055}}
```

```
In[5]:= X = x /. S[[2]]
```

```
Out[5]= 2.62055
```

```
In[6]:= (ω + Tanh[Cos[2 π - X]] == 0)
```

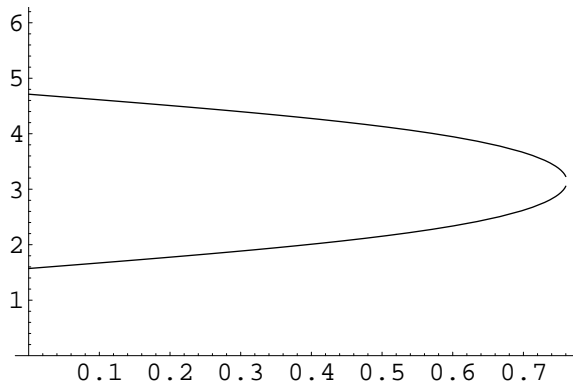
```
Out[6]= True
```

Mit Satz (1.6) kann man zeigen, dass der kleinere dieser beiden Werte x^* stets ein stabiler und der größere, x^{**} ein instabiler Gleichgewichtspunkt ist. Werden also der x^* oder x^{**} als Anfangswert für das Anfangswertproblem verwendet, so bleibt dessen Lösung

konstant in diesem Wert. Ist der gewählte Anfangswert ein anderer beliebiger Wert, so nähert sich die Lösung stets an x^* (modulo 2π) für $t \rightarrow \infty$ an. D.h. für Anfangswerte aus $[0, x^{**})$ strebt die Lösung direkt gegen x^* , für Werte $x(0) > x^{**}$ wird $x^* + 2\pi$ angenähert.

Die beiden Gleichgewichtswerte hängen dabei allein von der zentralen Frequenz ω ab.

```
In[1]:= ClearAll[ $\omega$ , A, x];
 $\omega$  = 0.76;
A = 1;
Plot[{Solve[x == ArcCos[(-ArcTanh[ $\omega$ )] / A], x][[1, 1, 2]],
      2  $\pi$  - Solve[x == ArcCos[(-ArcTanh[ $\omega$ )] / A], x][[1, 1, 2]]},
      { $\omega$ , 0, .76}, PlotRange -> {0, 2  $\pi$ }
```



```
Out[4]= - Graphics -
```

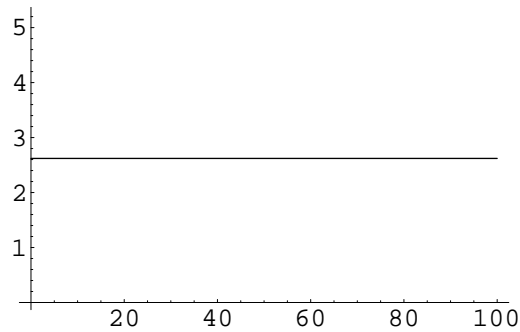
Setzt man also x^* als Anfangswert für die Differentialgleichung (2.9), so erhält man eine konstante Phase und damit auch eine konstante Ausgabe der Potentialfunktion $V(x)$, das Ruhepotential.

Mit *Mathematica* wird dies anschließend mit einem so gewählten Wert für ω , dass die Bedingung $\omega \in P(V(x)) = \tanh(A \cos(x))$ erfüllt ist, vorgeführt. Da der Faktor A ein Parameter für die Sensitivität der Potentialfunktion ist, und nur qualitativ das System verändert, soll A im folgenden Eins gesetzt werden.

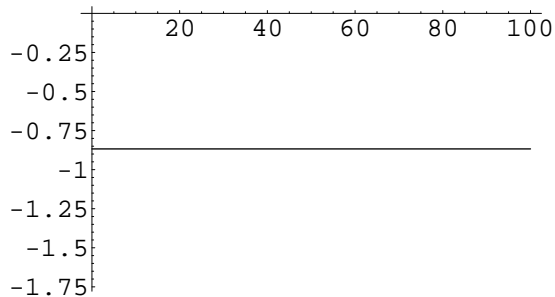
```
In[1]:=  $\omega$  = .7;
A = 1;
S = Solve[x[t] == ArcCos[(-ArcTanh[ $\omega$ )] / A], x[t]];
a = x[t] /. S[[1]]
```

```
Out[4]= 2.62055
```

```
In[5]:= ClearAll[AA, EE, solution];
AA = 0;
EE = 100;
solution = NDSolve[{x'[t] == ω + Tanh[A Cos[x[t]]], x[0] == a}, x, {t, AA, EE}];
Plot[x[t] /. solution, {t, AA, EE}]
Plot[A Cos[(x[t] /. solution)], {t, AA, EE}]
```



Out[9]= - Graphics -



Out[10]= - Graphics -

Wie die *Mathematica*-Ausgaben zu erkennen geben, verhält sich das VCON-Modell für den Fall eines Neurons im Ruhezustand qualitativ wie ein reales Neuron. Die genaue Skalierung der Werte wird hier nicht berücksichtigt.

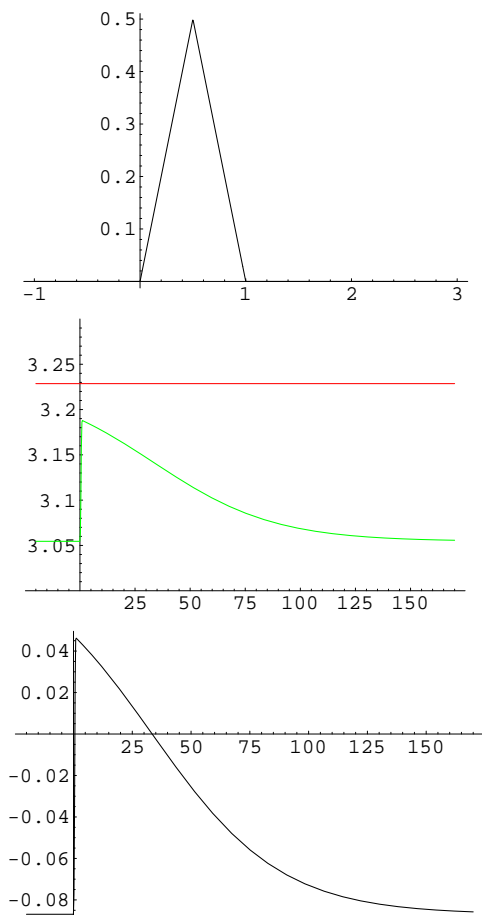
2. Ein Neuron unter dem Einfluß eines einmaligen Erregungsimpulses

Es wird nun von einer einzelnen vorgeschalteten Zelle ausgegangen, welche durch eine singuläre Potentialänderung und somit ggf. Transmitterausschüttungen auf das Potential der nachgeschalteten Zelle einwirkt. Als Potentialfunktion der ersten Zelle $V(y(t))$ wird in diesem Fall eine Zeltfunktion angenommen. Je größer deren Integral ist, desto stärker der Impuls.

Als Anfangswert soll wieder der stabile Gleichgewichtspunkt der Gleichung (2.13) gesetzt werden. Auf Grund der Abhängigkeit von der zentralen Frequenz und deren Eigenschaft $\omega \in [0, \tanh(\cos(z))]$, für ein $z \in \mathbb{R}$, kann dieser Wert nur zwischen $x = \arccos(-\frac{\arctanh(0)}{A}) = \frac{\pi}{2}$ und $x = \arccos(-\frac{\arctanh(\tanh(1))}{A}) = \pi$ liegen. Damit nun, ausgehend vom Ruhepunkt, das Potentialverhalten der Zelle qualitativ entsprechend modelliert werden kann, wird eine Phasenverschiebung um $\frac{\pi}{2}$ vorgenommen. Außerdem muß ω so gewählt werden, dass sich x_0 und x_1 um weniger als $\frac{\pi}{2}$ unterscheiden. Dadurch

wird gewährleistet, dass sich bei einem exzitatorischen Erregungsimpuls das Potential in positiver Richtung ändert, da sich das Ruhepotential bereits im ansteigenden Phasenbereich des Cosinus befindet. Das Potential des Neurons soll also fortan unter den obigen Bedingungen, in der Form $V(x(t) + \frac{\pi}{2})$ betrachtet werden.

```
In[1]:= Needs["Graphics`Colors`"]
In[2]:= ClearAll[t1, gmax, ω, A, solution, s, a];
t1 = 1;
gmax = 1;
g[t_] := If[t < 0, 0,
  If[0 ≤ t < t1/2, gmax t,
    If[t1/2 ≤ t < t1, gmax (t1 - t),
      If[t ≥ t1, 0]]]]
ω = .76;
A = 1;
s = Solve[s == ArcCos[(-ArcTanh[ω]) / A], s];
a = s /. S[1];
solution = NDSolve[{x'[t] == ω + Tanh[A Cos[x[t]] + g[t]], x[0] == a}, x, {t, -20, 170}];
Plot[g[t], {t, -1, 3}];
Plot[{x[t] /. solution, 2 π - a}, {t, -20, 170},
  PlotStyle → {Green, Red}, PlotRange → {3, 3.3}];
Plot[A Cos[x[t] /. solution] + π/2, {t, -20, 170}];
```

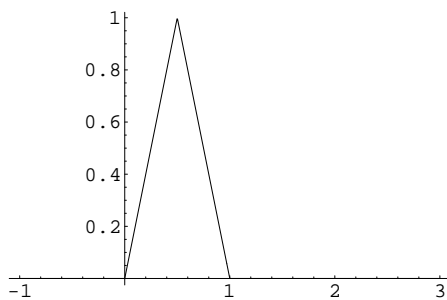


Eine kurzzeitige Potentialerhöhung, ausgedrückt durch die Zeltfunktion $g(t)$ bewirkt ein Ansteigen der Phase $x(t)$. Jedoch wird der kritische Wert x^{**} - hier in der zweiten Graphik rot eingezeichnet - nicht erreicht. Deshalb bewegt sich die Phase nach Been-

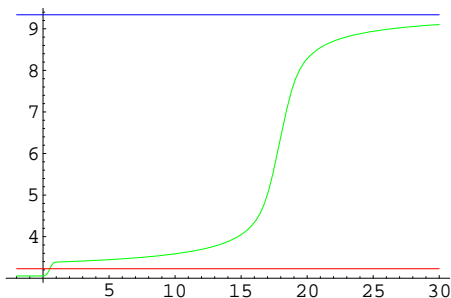
digung des Erregungsimpulses wieder auf den stabilen Gleichgewichtswert x^* zu. Dies bewirkt wiederum nur ein rasches Ansteigen der Potentialfunktion und dann ein monotonen Abfallen bis der Gleichgewichtspunkt wieder erreicht ist.

Ist der Impuls jedoch groß genug, dass durch ein starkes Ansteigen der Phase der instabile Gleichgewichtswert x^{**} - hier wieder rot merkiert - überschritten wird, so strebt die Phase nach dem Auslaufen der externen Erregung gegen $x^* + 2\pi$, hier in der zweiten Graphik die blaue Linie. Bei diesem Prozeß durchläuft somit die Phase, ursprünglich ausgehend von x^* bis $x^* + 2\pi$ eine vollständige Periode der Potentialfunktion, des Cosinus. Dies hat das hier zu beobachtende Verhalten der Funktion zur Folge: das Potential steigt zu einem positiven Extremum an, fällt anschließend ab zu einem negativen Tiefpunkt und bewegt sich schließlich wieder auf den Ausgangswert x^* zu.

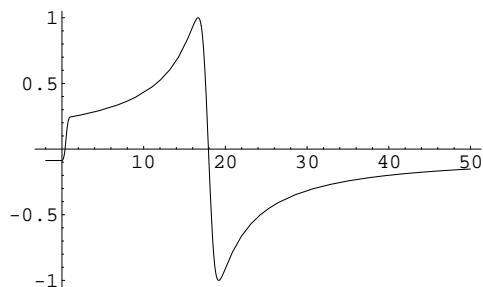
```
In[14]:= ClearAll[t1, gmax, solution];
t1 = 1;
gmax = 2;
solution = NDSolve[{x'[t] ==  $\omega$  + Tanh[A Cos[x[t]] + g[t]], x[0] == a}, x, {t, -3, 50}];
Plot[g[t], {t, -1, 3}]
Plot[{x[t] /. solution, 2  $\pi$  - a, a + 2  $\pi$ }, {t, -2, 30}, PlotStyle -> {Green, Red, Blue}]
Plot[A Cos[(x[t] /. solution) +  $\pi$  / 2], {t, -2, 50}]
```



Out[18]= - Graphics -



Out[19]= - Graphics -



Out[20]= - Graphics -

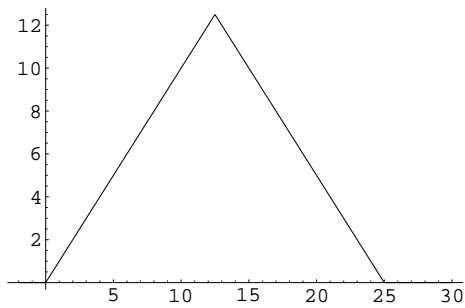
Wie in dieser letzten *Mathematica*-Ausgabe deutlich wird, hat die modellierte Poten-

tialfunktion bereits große Ähnlichkeit mit dem Aktionspotential einer Zelle, samt anschließender Refraktärphase.

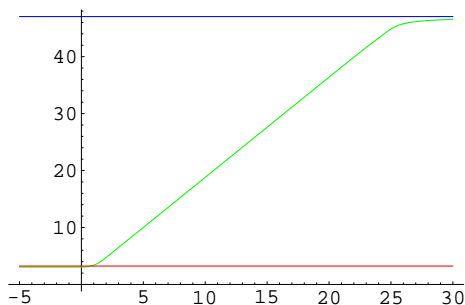
Bei sehr starken oder lang andauernden Reizen ergeben sich somit genauso viele Hoch- und Tiefpunkte, wie oft die Phase 2π durchläuft. Nach Ende des Stimulus nähert sich $x(t)$ dann an genau dieses Vielfache von $x^* + 2\pi$ an.

Im nachfolgenden Beispiel steigt die Phase durch den Erregungsinput 25 Zeiteinheiten lang an und strebt anschließend gegen $x^* + 7 \cdot 2\pi$. In der Potentialfunktion äußert sich dies durch sieben ausgeführte Aktionspotentiale.

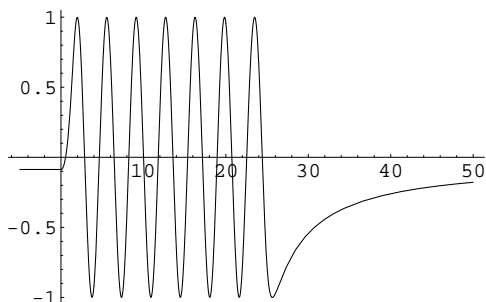
```
In[14]:= ClearAll[t1, gmax, solution];
t1 = 25;
gmax = 1;
solution = NDSolve[{x'[t] ==  $\omega + \text{Tanh}[A \text{Cos}[x[t]] + g[t]$ ], x[0] == a}, x, {t, -5, 50}];
Plot[g[t], {t, -2, 30}]
Plot[{x[t] /. solution,  $2\pi - a$ ,  $7(2\pi) + a$ },
{t, -5, 30}, PlotStyle -> {Green, Red, Blue}]
Plot[A Cos[(x[t] /. solution) +  $\pi/2$ ], {t, -5, 50}, PlotRange -> All]
```



Out[18]= - Graphics -



Out[19]= - Graphics -

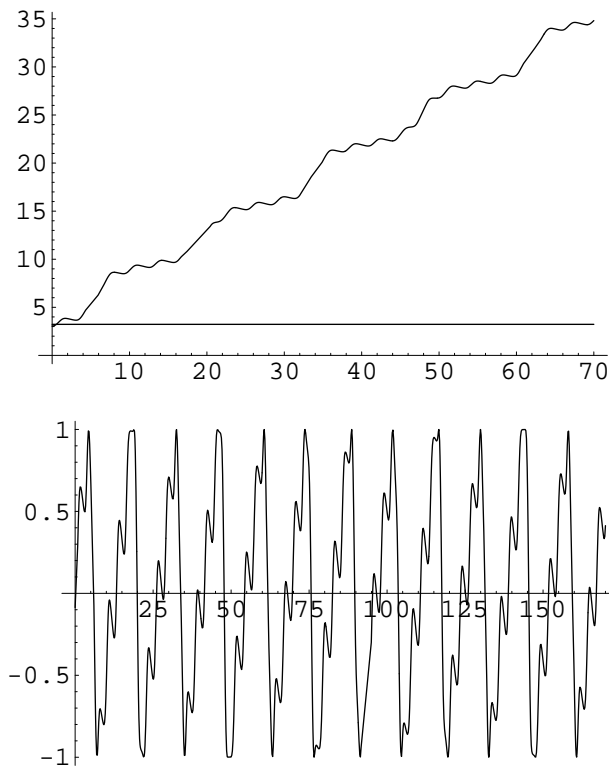


Out[20]= - Graphics -

3. Oszillierende Erregungen des Neurons

Wird ein Neuron nun überschwellig erregt, so feuert es Aktionspotentiale und gibt diese als Erregungsmuster an alle nachgeschalteten Neurone weiter. Soll also nun dieses Szenario mit dem Modell beschrieben werden, so muss man jetzt als Potentialfunktionen periodische Funktionen wählen. Im folgenden Beispiel wurde dafür $g(t) = \sin(2t)$ gesetzt.

```
In[1]:= ClearAll[ $\omega$ , A, solution, S, a];
g[t_] := Sin[2 t];
 $\omega$  = .76;
A = 1;
S = Solve[s == ArcCos[(-ArcTanh[ $\omega$ )] / A], s];
a = s /. S[[1]];
solution = NDSolve[{x'[t] ==  $\omega$  + Tanh[A Cos[x[t]] + g[t]], x[0] == a}, x, {t, -20, 170}];
Plot[{x[t] /. solution, 2  $\pi$  - a}, {t, 0, 70}];
Plot[A Cos[(x[t] /. solution) +  $\pi$  / 2], {t, 0, 170}];
Null
```



Sowohl die Phase als auch die Potentialfunktion scheinen gewisse Regelmäßigkeiten in ihrem Verhalten aufzuweisen. In welcher Weise dies genau geschieht kann man mit verschiedenen analytischen Methoden untersuchen.

Phase locking - Eigenschaft des VCON

Das hier vorgestellte Grundmodell des VCON kann nun so verwendet werden, dass diese einfachen Prozesse, hier demonstriert an einem Neuron, in einen größeren Zusam-

menhang gestellt werden. Das Modell eines Neurons mit Einfluss eines vorgeschalteten Neurons wird dann so umgeformt, dass eine richtige Rückkopplungsschleife entsteht. Mit einem *Phasendetektor* wird dann pro Schleifendurchlauf der Unterschied zwischen der Inputphase, und der Phase des Neurons überprüft und eine leichte Veränderung der Phase des Neurons vorgenommen. Betrachtet man dann das Langzeitverhalten der Phasen und insbesondere ihr Verhältnis zueinander, so kann man feststellen, dass gewisse *Phase-Locking* Eigenschaften auftreten. D.h. das Verhältnis der Phasen schwingt sich auf einen bestimmten Wert ein und behält diesen. Solche Synchronisationen der Phasen können dann mit speziellen Methoden explizit berechnet werden.

2.2.2 Neuronen-Netzwerke

Das Modell-Prinzip des VCON kann dann ohne weiteres auf größere Neuronen-Netzwerke ausgeweitet werden. Allerdings erhöht sich die Komplexität des System damit ungemein und erfordert hohe Rechenleistungen.

In der Gehirnforschung sind diese Ergebnisse aber natürlich von größtem Interesse, da man ständig auf der Suche nach Prinzipien ist, welche das Verständnis der Vorgänge im gewaltigen Neuronen-Netzwerk des Gehirns erleichtern. So ist gerade die Untersuchung von Synchronisationsprozesse im Gehirn bei bestimmten Tätigkeiten ein aktueller Forschungsbrennpunkt in der Neurowissenschaft.

2.3 Andere Neuronen-Modelle

Das hier vorgestellte VCON-Modell von F.C. Hoppensteadt ist dabei nur eines unter vielen Neuronen-Modellen. Jeder der Ansätze betont dabei einen etwas anderen Aspekt des zu modellierenden Phänomens.

In den 1950er Jahren stellten Hogkin und Huxley das erste Standard-Modell eines Aktionspotentiale feuernenden Neurons auf. Kurz darauf entwickelten FitzHugh und Nagumo das mathematische Modell zu diesem Ansatz. Seitdem wurden viele verschiedene Neuronen-Modelle präsentiert, was die große Vielfalt der Herangehensweisen an eine Modellierungs-Problematik gut demonstriert. Das VCON-Modell von Hoppensteadt wurde 1986 erstmals veröffentlicht.

Verwendete Literatur

Hauptliteraturquellen dieses Kapitels waren [HOP] und [HOP/PES]. Außerdem wurden auch für die Abbildungen die Quellen [YSH] und [WWW] verwendet.

Kapitel 3

Anwendungen von Markov-Ketten in der Biologie

Bei der Modellierung biologischer Phänomene sind viele Mechanismen unbekannt und zu komplex, um sie deterministisch zu modellieren. Deshalb werden sie durch Zufallsgrößen beschrieben, so dass verschiedene stochastische Methoden angewandt werden können.

Ein mathematisches Modell, welches dies leistet und in diesem Kapitel vorgestellt werden soll, ist die Theorie von Markov-Ketten, ein diskreter stochastischer Prozeß mit interessanten Eigenschaften und Anwendungen in der Biologie.

3.1 Einführung in die Theorie der Markov-Ketten

Definition 3.1 Eine **Markov-Kette** ist eine Folge von Zufallsvariablen $(X_i)_{i \in \mathbb{N}}$ auf einem Wahrscheinlichkeitsraum (Ω, \mathcal{F}, P) mit Werten in einem diskreten, höchstens abzählbaren, meßbaren Raum $(E, \mathcal{P}(E))$ mit $E \neq \emptyset$, falls für alle $x_0, \dots, x_{n+1} \in E$, $n \in \mathbb{N}$ die folgende **Markov-Eigenschaft** gilt:

$$P(X_{n+1} = x^{n+1} | X_0 = x^0, \dots, X_n = x^n) = P(X_{n+1} = x^{n+1} | X_n = x^n) \quad (3.1)$$

E wird dabei als **Zustandsraum** bezeichnet und die Verteilung der ersten Zufallsvariablen X_0 heißt **Startverteilung** der Markov-Kette. Die bedingte Einzelwahrscheinlichkeit, im $(n+1)$ ten Schritt von $X_n = x_n$ nach $X_{n+1} = x_{n+1}$ überzugehen, sei folgendermaßen definiert

$$P(X_{n+1} = x^{n+1} | X_n = x^n) =: \Pi(x^n, x^{n+1}) =: p_{i,j}.$$

Sie werden in der **Übergangsmatrix**

$$\Pi_n := (\Pi(x_i, x_j))_{x_i, x_j \in E}$$

als **Übergangswahrscheinlichkeiten** zusammengefasst.

Hängt die Übergangsmatrix Π nicht von $n \in \mathbb{N}_0$ ab, so wird die zugehörige Markov-Kette **homogen** genannt.

Die Markov-Eigenschaft (3.1) besagt also, dass die Realisierung der $(n+1)$ ten Zufallsvariablen ausschließlich vom Ergebnis der unmittelbar vorhergegangenen Zufallsvariablen X_n abhängt, alle früheren Ergebnisse der Zufallsvariablen sind unwichtig. Eine Markov-Kette ist daher ein stochastischer Prozess mit 'Kurzzeitgedächtnis'.

Eine Zufallsvariable der Markov-Kette nimmt jeweils einen Wert aus dem Zustandsraum E , also einen bestimmten Zustand an. Seien $x_i, x_j \in E$ zwei Zustände, so beschreibt die Übergangswahrscheinlichkeit p_{ij} die Wahrscheinlichkeit, dass die Zufallsvariable X_{n+1} den Zustand x_i annimmt, unter der Voraussetzung, die Zufallsvariable X_n hatte als angenommenen Zustand den Wert x_j .

Da die Eigenschaft (3.1) für alle $n \in \mathbb{N}$ gelten soll, sind diese Übergangswahrscheinlichkeiten bei allen Übergängen von einer Zufallsvariablen zur anderen gleich und können in eine einzige Übergangsmatrix eingetragen werden. Diese Matrix hat dabei die spezielle Form einer stochastischen Matrix.

Definition 3.2 Eine $(n \times n)$ -Matrix $(p_{ij})_{i,j=1,\dots,n}$ heißt **stochastisch**, wenn gilt

$$\sum_{j=1}^n p_{ij} = 1 \quad \forall i,$$

d. h. alle Zeilensummen der Matrix müssen 1 ergeben.

Eine Markov-Kette ist durch eine feste Anfangsverteilung und die Matrix der Übergangswahrscheinlichkeiten Π eindeutig bestimmt. Auf Grund des Multiplikationssatzes für bedingte Wahrscheinlichkeiten und der Markov-Eigenschaft gilt:

$$\begin{aligned} &P(X_0 = x^0, X_1 = x^1, \dots, X_n = x^n = i) \\ &= P(X_0 = x^0) P(X_1 = x^1 | X_0 = x^0) P(X_2 = x^2 | X_0 = x^0, X_1 = x^1) \cdot \dots \\ &\quad \cdot P(X_n = x^n | X_0 = x^0, \dots, X_{n-1} = x^{n-1}) \\ &\stackrel{ME}{=} p_0 \cdot p_{0,1} \cdot p_{1,2} \cdot \dots \cdot p_{n-1,n} \end{aligned}$$

Die Wahrscheinlichkeit, dass die Zufallsvariable X_n bei einer Anfangsverteilung π den Zustand $i \in E$ annimmt, berechnet sich dann mit Hilfe der Formel der vollständigen Wahrscheinlichkeit folgendermaßen durch Induktion. π sei im Folgenden stets ein Zeilenvektor.

Induktionsanfang:

$$\begin{aligned} P^\pi(X_1 = x_i) &= \sum_{x_j \in E} P^\pi(X_1 = i | X_0 = x_j) P^\pi(X_0 = x_j) \\ &= \sum_{x_j \in E} p_{ji} \pi_j = \sum_{j \in E} \pi_j p_{ji} \\ &= (\pi \Pi)_i \end{aligned}$$

Induktionsannahme: $P^\pi(X_{n-1} = x_j) = (\pi\Pi^{n-1})_j$

Induktionsschluß:

$$\begin{aligned} P^\pi(X_n = x_i) &= \sum_{j \in E} P^\pi(X_n = x_i | X_{n-1} = x_j) P^\pi(X_{n-1} = x_j) \\ &\stackrel{IV}{=} \sum_{j \in E} p_{ji} (\pi\Pi^{n-1}) = (\pi\Pi^{n-1}) \sum_{j \in E} p_{ji} \\ &= (\pi\Pi^n)_i \end{aligned}$$

Die allgemeine Wahrscheinlichkeit, dass die Markov-Kette zum Zeitpunkt n im Zustand x_i ist, wenn sie zum Zeitpunkt k in x_j war, wird mit der so genannten *Chapman-Kolmogorov-Gleichung* berechnet:

Satz 3.3 (Chapman-Kolmogorov-Gleichung)

Für eine Markov-Kette $(X_n)_{n \geq 0}$ und $k, n, m \in \mathbb{N}$ mit $k < n < m$ gilt

$$P(X_n = x_j | X_k = x_i) = \sum_{i \in E} P(X_n = x_j | X_m = x_i) P(X_m = x_i | X_k = x_i), \quad \forall x_h, x_j \in E.$$

Definition 3.4 Sei $(X_n)_{n \geq 0}$ eine Markov-Kette mit Übergangsmatrix Π und $x \in E$ ein Zustand aus dem zugehörigen Zustandsraum E der Markov-Kette. Gilt

$$\Pi_{(x,x)} = 1,$$

so heißt x **absorbierender Zustand**.

Nimmt also eine Markov-Kette einmal einen absorbierenden Zustand an, so kann sie diesen mit Wahrscheinlichkeit eins nicht mehr verlassen.

Analog zur Dynamik von Differentialgleichungen ist hier wiederum interessant, ob sich das Verhalten der Markov-Kette mit der Zeit immer weniger ändert, d.h. in eine gleichbleibende Wahrscheinlichkeitsverteilung, eine so genannte *stationäre Verteilung* einpendelt, oder ob sich diese ständing weiter stark verändern wird.

Definition 3.5 $(X_n)_{n \geq 0}$ sei eine Markov-Kette mit Übergangsmatrix Π . Die zu einer Dichte π gehörige Verteilung auf E wird **stationäre Verteilung** genannt, falls für diese Dichte

$$\pi\Pi = \pi$$

gilt.

Da die Dichte und die Verteilung wegen ihrer eindeutigen Zuordnung im diskreten Fall miteinander im Sprachgebrauch identifiziert werden können, wird auch π als stationäre Verteilung bezeichnet.

Der *Ergodensatz* für Markov-Ketten gibt genauere Auskunft über das Verhältnis von Markov-Ketten und möglichen zugehörigen stationären Verteilungen.

Satz 3.6 (Ergodensatz für Markov-Ketten mit endlichem Zustandsraum)
 $(X_n)_{n \geq 0}$ sei eine Markov-Kette mit Übergangsmatrix Π und endlichem Zustandsraum $E = \{x_1, \dots, x_r\}$. Gibt es ein $m \in \mathbb{N}$, so dass alle Elemente von Π^m echt positiv sind, dann existiert der Limes

$$K = \lim_{n \rightarrow \infty} \Pi^n.$$

Dabei sind die Zeilen von K alle gleich und stellen jeweils die einzige stationäre Anfangsverteilung κ der Markov-Kette dar.

Aus dem Ergodensatz lässt sich außerdem ableiten, dass eine Markov-Kette unter den Bedingungen von Satz 3.6 bei verschiedenen Anfangsverteilungen stets nach einiger Zeit stationär wird, d.h. von der stationären Verteilung nur noch um ein beliebig kleines ϵ abweicht.

Eine weitere Fragestellung von Markov-Ketten ist die Frage, ob einzelne Zustände von der Markov-Kette angenommen werden, und wann oder ob diese dann noch einmal auftreten. Hierzu noch einige weitere Definitionen:

Definition 3.7 Gegeben ist eine Markov-Kette $(X_n)_{n \in \mathbb{N}_0}$ mit Zustandsraum E . Für einen Zustand $x \in E$ wird mit

$$\tau_x = \inf(n \in \mathbb{N} : X_n = x)$$

die **Eintrittszeit** in den Zustand x bezeichnet. War x bereits der Startwert von X_0 , so heißt τ_x **Rückkehrzeit**.

Definition 3.8 Die Wahrscheinlichkeit, nach einem Start in x wieder in endlicher Zeit dorthin zurück zu kehren wird mit

$$F_1(x, x) = P(\tau_x < \infty)$$

angegeben. Wie wahrscheinlich es ist, unendlich-mal wieder zu x zurück zu kommen bezeichnet

$$F_\infty(x, x) = P(X_n = x \text{ für unendlich viele } n),$$

und schließlich wird der Erwartungswert der Rückkehrten nach x durch

$$G(x, x) = \sum_{n \geq 0} P(X_n = x) = E^x\left(\sum_{n \geq 0} 1_{X_n=x}\right)$$

beschrieben.

Gilt $F_1(x, x) = 1$, so heißt x **rekurrent**, gilt dagegen $F_1(x, x) < 1$ so wird x als **transient** bezeichnet.

Lemma 3.9 Ist ein Zustand x rekurrent, so gilt $F_\infty(x, x) = 1$ und $G(x, x) = \infty$. Ist x dagegen transient, so folgt $F_\infty(x, x) = 0$ und $G(x, x) < \infty$.

Man kann sich auch noch für die Dauer bis zur nächsten Rückkehr eines Zustandes interessieren.

Definition 3.10 Sei $x \in E$ ein rekurrenter Zustand. Gilt $E^x(\tau_x) < \infty$, d.h. die erwartete Rückkehrzeit ist endlich, so heißt x **positiv rekurrent**.

Ist ein rekurrenter Zustand nicht positiv rekurrent, so heißt er **nullrekurrent**.

Im nullrekurrenten Fall kommt die Markov-Kette $(X_n(\omega))$ für fast alle $\omega \in \Omega$ wieder nach x zurück, aber der Erwartungswert als eine Art Mittelwert über Ω ist nicht endlich.

Satz 3.11 (Positive Rekurrenz und stationäre Verteilung)

Sei $(X_n)_{n \geq 0}$ eine Markov-Kette mit Startwert $P(X_0 = x_i) = 1$ für einen rekurrenten Zustand $x_i \in E$, dann gilt:

1. Ist x_i sogar positiv rekurrent, dann gilt, es existiert eine stationäre Verteilung π mit

$$\pi_i = \frac{1}{E^{x_i}(\tau_{x_i})} > 0.$$

2. Existiert hingegen eine stationäre Verteilung π mit $\pi_i > 0 \forall i$ so folgt positive Rekurrenz für x_i .

Nach diesem Satz ist also die Existenz einer stationären Verteilung äquivalent zur positiven Rekurrenz aller rekurrenten Zustände.

3.2 Simulation und Anwendungen des Modells

Nach dieser theoretischen Einführung soll nun eine Simulation einer Markov-Kette mit *Mathematica* vorgestellt und einige Anwendungen des Modells in der Biologie angeführt werden. Dies sind zunächst das noch etwas abstraktere Beispiel des *Random-walk* in den ersten Dimensionen, anschließend das *Ehrenfest-Modell*, welches den Austausch von Molekülen in zwei abgetrennten Kammern beschreibt und schließlich ein Beispiel aus der *Genetik*, wobei das zufällige Auswahlverfahren bei der Realisierung von Genen und dessen Auswirkungen modelliert werden soll.

3.2.1 Simulation einer Markov-Kette

Mit der Eingabe einer Matrix, einem Anfangswert und einer Schrittzahl für die Länge der Simulation läßt sich mit der folgenden Funktion `MarkovSimulation`, unter der Ausnutzung der Gleichverteilung der Ergebnisse der *Mathematica*-Funktion `Random` eine Realisierung einer Markov-Kette durchführen.

```

In[1]:= MarkovSimulation[Übergangsmatrix_,
  Anfangswert_, Schritte_, Frage1_, Frage2_] := Module[
  {M = Übergangsmatrix, x = Anfangswert, s = Schritte,
    d, y, W, v, w, S, F1 = Frage1, F2 = Frage2},

  (*****Überprüfung der Eingabe*****)
  If[!(IntegerQ[s] && IntegerQ[x] && (s > 0) && (x > 0)),
    Return["Anfangswert und Schritte müssen natürliche Zahlen sein!"]];
  If[Length[Dimensions[M]] ≠ 2, Return["Keine Matrix!"]];
  If[Dimensions[M][[1]] ≠ Dimensions[M][[2]],
    Return["Die Matrix ist nicht quadratisch!"]];
  d = Length[M];
  If[x0 > d, Return["Der Anfangswert ist zu groß!"]];
  For[i = 1, i <= d, i++,
    For[j = 1, j <= d, j++,
      If[M[[i, j]] < 0,
        Return["Es gibt einen negativen Eintrag in der Matrix!"]]]
  ];
  For[i = 1, i <= d, i++,
    y = 0;
    For[j = 1, j <= d, j++,
      y = y + M[[i, j]]];
    If[y ≠ 1,
      Return["Dies ist keine stochastische Matrix!"]]];

  (*****KummulativeEntscheidungsMatrix*****)
  K = IdentityMatrix[d];
  For[i = 1, i <= d, i++,
    For[j = 1, j <= d, j++,
      K[[i, j]] = Sum[M[[i, 1]], {1, 1, j}]]
  ];

  (*****Simulation*****)
  W = {};
  W = Append[W, x];
  For[k = 1, k <= s, k++,
    Clear[v];
    v = Random[];
    For[l = 1, l <= d, l++,
      If[v <= K[[x, l]],
        x = l;
        W = Append[W, x];
        Break[]];
  ];
  ];
  (*Frage1: Soll Übergangsmatrix gezeigt werden?
    Frage2: Soll die Werteliste der Simulation ausgegeben werden?*)
  If[F1 == 1,
    Print[M // MatrixForm]];
  If[F2 == 1, Print[W]];
  W
  ];

```

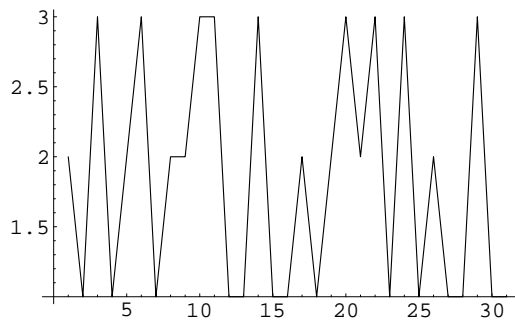
Zu Beginn des Algorithmus wird überprüft, ob die eingegebene Matrix einer stochastischen Matrix entspricht und ob die Angaben des Startwerts und der Schrittzahl vernünftig sind. Anschließend wird die Matrix so transformiert, dass in jeder Zeile die

Einträge von links nach rechts aufsummiert werden und die Matrix somit eine Form mit kummulativen Verteilungen in jeder Zeile erhält. In jeder Zeile der Matrix wird also das Intervall $[0, 1]$ in d Teile unterteilt, wobei d die Dimension der Matrix und damit die Anzahl der Elemente des Zustandraums ist.

Der eigentliche Simulationsschritt geschieht nun dadurch, dass in der Zeile des Startwerts beginnend, eine Realisierung von `Random`, welche eine Zufallszahl zwischen Null und Eins ausgibt verwendet wird. Es wird dann überprüft, in welchen Unterabschnitt von $[0, 1]$ die Zufallszahl trifft. Der entsprechende Eintrag, welcher diesen Abschnitt nach oben begrenzt wird dann als nächste Realisierung der Markov-Kette interpretiert. Nun wird der nächste Schritt, in der Zeile des realisierten Zustands auf die gleiche Weise mit einer Zufallszahl von `Random` durchgeführt.

Die Realisierungen der Markov-Kette werden, beim Startwert beginnend nacheinander in eine Liste geschrieben. Zuletzt wird noch die Ausgabe der Funktion durch die Fragen `F1` und `F2` geregelt.

```
In[2]:= MarkovPlot[Übergangsmatrix_, Anfangswert_, Schritte_, Frage1_, Frage2_] := Module[
  {M = Übergangsmatrix, x = Anfangswert, s = Schritte, F1 = Frage1, F2 = Frage2, d, S},
  ListPlot[MarkovSimulation[M, x, s, F1, F2], PlotJoined -> True];
  Return[]
]
MarkovPlot[{{.3, .3, .4}, {.5, .1, .4}, {.8, .1, .1}}, 2, 30, 1, 1]
(
  0.3 0.3 0.4
  0.5 0.1 0.4
  0.8 0.1 0.1
)
{2, 1, 3, 1, 2, 3, 1, 2, 2, 3, 3, 1, 1, 3, 1, 1, 2, 1, 2, 3, 2, 3, 1, 3, 1, 2, 1, 1, 3, 1, 1}
```



Die Funktion `MarkovPlot` ruft dann `MarkovSimulation` auf, produziert eine Graphik der realisierten Markov-Kette und gibt optional die Übergangsmatrix und die Realisierungen in einer Liste aus, wie das obige Beispiel einer 3×3 -Matrix mit 30 Simulationsschritten zeigt.

Die folgende Funktion `RandomMarkov` erleichtert für umfangreichere Beispiele die Angabe einer stochastischen Matrix.

```

In[4]:= RandomMarkov[Dimension_] := Module[
  {d = Dimension, M, s},
  M = IdentityMatrix[d];
  For[i = 1, i <= d, i++,
    s = 0;
    For[j = 1, j <= d, j++,
      M[[i, j]] = Random[];
      s = s + M[[i, j]];];
  M[[i]] = M[[i]] / s;];
  M
]
RandomMarkov[5] // MatrixForm

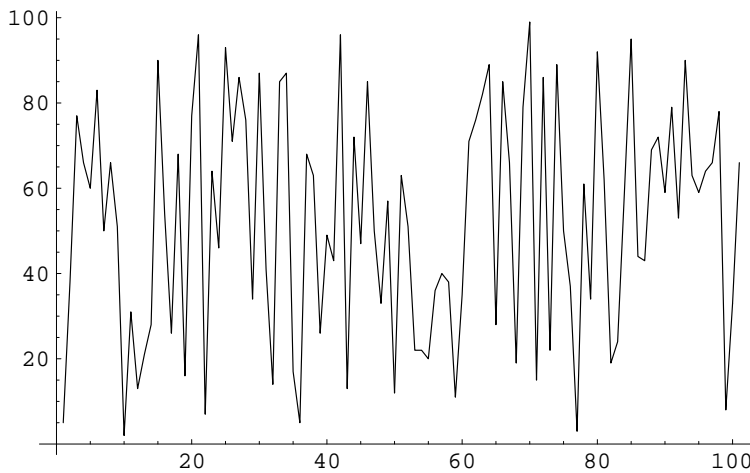
Out[5]//MatrixForm=

$$\begin{pmatrix} 0.0902007 & 0.0899132 & 0.384625 & 0.397151 & 0.0381094 \\ 0.294339 & 0.072893 & 0.197237 & 0.229537 & 0.205993 \\ 0.0801224 & 0.23348 & 0.26428 & 0.162491 & 0.259628 \\ 0.033462 & 0.302173 & 0.276092 & 0.225603 & 0.16267 \\ 0.115955 & 0.146353 & 0.361446 & 0.357651 & 0.0185956 \end{pmatrix}$$


```

Nun können Markov-Ketten auch mit größeren Zustandsräumen leicht simuliert werden.

```
In[6]:= MarkovPlot[RandomMarkov[100], 5, 100, 0, 0]
```



3.2.2 Random-Walks

Zufallsbewegungen von Teilchen jeglicher Art spielen u.a. in der Molekularbiologie eine wichtige Rolle. Hier sollen nun Random Walks in Form einer symmetrischen Irrfahrt auf \mathbb{Z}^d betrachtet werden. Das beobachtete Teilchen kann sich also auf einem gitterartigen Gebilde frei bewegen, und pro Zeitschritt von einem Zustandspunkt in den nächsten springen.

Eine symmetrische Irrfahrt kann durch eine Markov-Kette mit dem Zustandsraum $E = \mathbb{Z}^d$ und der Übergangsmatrix $\Pi = (p_{ij})_{i,j \in E}$ dargestellt werden, wobei die Übergangswahrscheinlichkeiten folgendermaßen aussehen:

$$p_{ij} = \begin{cases} \frac{1}{2d} & \text{falls } |i - j| = 1, \\ 0 & \text{sonst.} \end{cases}$$

Dieser Fall wird **symmetrische Irrfahrt** genannt, da es gleich wahrscheinlich ist, von einem Zustand ausgehend in die möglichen angrenzenden Zustände überzugehen. Das

bedeutet, die Übergangswahrscheinlichkeiten für die nächsten anzunehmenden Zustände sind gleichverteilt. Außerdem wird festgelegt, dass die Wahrscheinlichkeit, in einem Zustand zu bleiben, gleich Null ist. Das Teilchen bewegt sich also in jedem Zeitschritt weiter. Natürlich kann man auch Irrfahrten mit anderen Verteilungen betrachten, aber da hier ein zufälliges Verhalten modelliert werden soll, wird im Folgenden nur die symmetrische Irrfahrt untersucht.

Nacheinander werden die Fälle der Dimensionen $d = 1, 2, 3$ behandelt. Man interessiert sich dabei jeweils dafür, ob ein Ausgangspunkt noch einmal erreicht wird, und wie lange dies im Mittel dauert. Allgemein ist es also die Frage nach dem Rekurrenztyp der Punkte des Gitters \mathbb{Z}^d .

Symmetrische Irrfahrt auf dem eindimensionalen Gitter \mathbb{Z} ($d = 1$)

Da der Zustandsraum E hier auf eine Dimension beschränkt ist, ergeben sich von einem Zustand aus nur zwei mögliche Bewegungsrichtungen. Die Übergangswahrscheinlichkeiten sind damit einfach $P^x(X_n = x + 1) = \frac{1}{2}$ und $P^x(X_n = x - 1) = \frac{1}{2}$, für $x \in \mathbb{Z}$. Auf Grund der Dimension $d = 1$ kann der Ausgangspunkt $x \in \mathbb{Z}$ nur in einer geraden Anzahl von Schritten $2n$ wieder erreicht werden, und zwar genau dann, wenn eine gleiche Anzahl n von Links- und Rechtsbewegungen ausgeführt worden ist. Somit ist die allgemeine Wahrscheinlichkeit nach $2n$ Schritten wieder zu x zurück zu kommen

$$P^x(X_{2n} = x) = \left(\frac{1}{2}\right)^{2n} \binom{2n}{n},$$

da $\binom{2n}{n}$ die Anzahl aller möglichen Schrittfolgen, bei Gesamtweglänge $2n$ und n Schritten in einer Richtung darstellt.

Um die Gitterpunkte $x \in \mathbb{Z}$ auf Rekurrenz oder Transienz zu testen, müssen nach Lemma 3.9 F_∞ oder G berechnet werden. Es gilt

$$G(x, x) = \sum_{n \geq 0} \Pi_{(x,x)}^{2n} = \sum_{n \geq 0} P^x(X_n = x) = \sum_{n \geq 0} \left(\frac{1}{2}\right)^{2n} \binom{2n}{n} = \infty,$$

da unter Zuhilfenahme der Stirling'schen Formel die Abschätzung $\left(\frac{1}{2}\right)^{2n} \binom{2n}{n} \sim \frac{1}{\sqrt{\pi n}}$

vorgenommen werden kann, und die Reihe $\sum_{n \geq 0} \frac{1}{\sqrt{\pi n}}$ divergiert.

Die erwartete Zeit bis zu einer Rückkehr in den Zustand x ist also unendlich und damit sind alle $x \in \mathbb{Z}$ rekurrent.

Man kann nun aber auch noch zeigen, dass in \mathbb{Z}^d , $d \leq 2$ alle Zustände nur nullrekurrent sind.

Beweis: Nach [GEO]. Angenommen, es existiert ein positiv rekurrenter Zustand $x \in \mathbb{Z}^d$, so gibt es nach Satz 3.11 eine stationäre Verteilung a . Als Verteilung besitzt a die übliche Eigenschaft

$$\sum_{y \in E} a_{(y)} = 1,$$

wobei $a_{(y)}$ die Wahrscheinlichkeit für den Zustand y gemäß der stationären Wahrscheinlichkeitsverteilung a der Markov-Kette darstellt. Auf Grund dieser Eigenschaft lässt sich also auch ein größter Wert $a_{(z_0)} = \max(a_{(y)}, y \in E)$ bestimmen.

Es gilt dann

$$m = a_{(z_0)} = (a\Pi)_{(z_0)} = \sum_{y \in E} \frac{1}{2d} 1_{\{|y-z_0|=1\}}(y) a_{(y)} = \frac{1}{2d} \sum_{y:|y-z_0|=1} a_{(y)}.$$

Da auf Grund der Beschaffenheit der symmetrischen Irrfahrt für alle Zustände y gilt, dass alle $a_{(y)}$ gleich sind und da durch die Bedingung $|y - z_0| = 1$ jeweils $2d$ Zustände in Frage kommen, folgt

$$m = \frac{1}{2d} \sum_{y:|y-z_0|=1} a_{(y)} = \frac{1}{2d} a_{(y)} 2d = a_{(y)}.$$

Dies bedeutet, dass alle Zustände y , welche der Bedingung $|y - z_0| = 1$ genügen das Maximum $m = a_{(y)}$ in der stationären Verteilung annehmen. Nimmt man nun ein neues $z_1 \in E$ aus der Menge der $y \in E$ mit $m = a_{(y)}$, so kann man wieder eine neue Menge von Zuständen $y \in E$ mit $|y - z_1| = 1$ und $m = a_{(y)}$ finden. Wird dieses Verfahren fortgeführt, so gilt schließlich $m = a_{(y)} \forall y \in \mathbb{Z}^d$. Dies steht dann aber im Widerspruch zu $\sum_{y \in \mathbb{Z}^d} a_{(y)} = 1$. Also können die Gitterpunkte von \mathbb{Z}^d nicht positiv rekurrent sein, und sind damit nullrekurrent. \square

In einem eindimensionalen Gitter ($d = 1$) kehrt die Markov-Kette also garantiert stets in den Startzustand wieder zurück, wann dies aber geschieht kann nicht vorhergesagt werden, d.h. dies geschieht nicht in endlicher Zeit.

Symmetrische Irrfahrt auf dem zweidimensionalen Gitter \mathbb{Z}^2 ($d = 2$)

Auf einem zweidimensionalen Gitter $E = \mathbb{Z}^2$ gibt es nun vier verschiedene Bewegungsrichtungen. Für die Einzelwahrscheinlichkeiten ergibt sich also in diesem Fall $p_{ij} = \frac{1}{4}$ für $|i - j| = 1$ und $p_{ij} = 0$ sonst. Um in einen Startpunkt wieder zurück zu kehren, müssen auch wieder jeweils die gleiche Anzahl von Links- und Rechtsbewegungen sowie Bewegungen nach oben und unten stattgefunden haben. Außerdem muß die Schrittzahl gerade sein. Wenn der Startwert also wieder erreicht werden soll, müssen insgesamt $2n$ Schritte ausgeführt werden. Man kann sich dabei k Schritte in die eine Dimensionsrichtung und $n - k$ in die andere bewegen. Für einen Weg der Länge $2n$ auf \mathbb{Z}^2 gibt es dann also $\sum_{k=0}^n \frac{(2n)!}{k!(n-k)!^2}$ Möglichkeiten und die Wahrscheinlichkeit einer Rückkehr

zum Zustand x berechnet sich somit folgendermaßen:

$$P^x(X_{2n} = x) = \left(\frac{1}{4}\right)^{2n} \sum_{k=0}^n \frac{(2n)!}{k!^2(n-k)!^2} \quad (3.2)$$

$$= \left(\frac{1}{4}\right)^{2n} \binom{2n}{n} \sum_{k=0}^n \binom{n}{k} \binom{n}{n-k} \quad (3.3)$$

$$= \left(\frac{1}{4}\right)^{2n} \binom{2n}{n}^2 \sum_{k=0}^n \frac{\binom{n}{k} \binom{n}{n-k}}{\binom{2n}{n}} \quad (3.4)$$

$$= \left(\frac{1}{4}\right)^{2n} \binom{2n}{n}^2 = \left(\left(\frac{1}{2}\right)^{2n} \binom{2n}{n}\right)^2 \quad (3.5)$$

Da in der vorletzten Zeile über die hypergeometrische Verteilung summiert wurde, und dieses bekanntlich Eins ergibt, fällt die gesamte Summe zum Faktor Eins zusammen. Wie im eindimensionalen Fall kann man nun mit derselben Abschätzung mit der Stirlingformel zeigen, dass alle Punkte des \mathbb{Z}^2 rekurrente Zustände sind:

$$G(x, x) = \sum_{n \geq 0} P^x(X_{2n} = x) = \sum_{n \geq 0} \left(\left(\frac{1}{2}\right)^{2n} \binom{2n}{n}\right)^2 \sim \sum_{n \geq 0} \left(\frac{1}{\sqrt{m}}\right)^2 = \sum_{n \geq 0} \frac{1}{m} = \infty$$

Untersucht man wieder die Art der Rekurrenz, so ergibt sich wie auf dem eindimensionalen Gitter, gemäß dem oben stehenden Beweis, für den Fall $d = 2$ Nullrekurrenz für alle Zustände des \mathbb{Z}^2 .

In einem zweidimensionalen Gitter ist also ebenfalls eine Rückkehr zum Ausgangspunkt sicher. Dies geschieht allerdings auch nicht in endlicher Zeit, wie bereits im vorangegangenen Beweis gezeigt wurde.

Diese Eigenschaft der Rekurrenz aller Zustände ist aber schon für \mathbb{Z}^3 und für alle Gitter höherer Dimensionen nicht mehr gegeben:

Symmetrische Irrfahrt auf höherdimensionalen Gittern \mathbb{Z}^d ($d \geq 3$)

In einem dreidimensionalen Gittergerüst sind die Bewegungsrichtungen des Teilchens wieder um zwei neue Richtungen, hinten und vorne erweitert und natürlich gilt allgemein, dass es in einem d -dimensionalen Gitter $2d$ Bewegungsrichtungen gibt. Wieder gilt für eine symmetrische Irrfahrt auf einem solchen Gitter, dass die Schrittzahl für eine Rückkehr zum Ausgangspunkt gerade sein muß, und dass sich die Anzahlen der Schritte in einer Richtung durch die Schrittzahl in der Gegenrichtung aufheben müssen.

Für eine Rückkehr zum Ausgangszustand ergibt sich die folgende Wahrscheinlichkeit:

$$P^x(X_{2n} = x) = \left(\frac{1}{2d}\right)^{2n} \sum_{\vec{k}} \frac{(2n)!}{k_1!^2 \cdots k_d!^2}$$

Dabei wird über alle $\vec{k} = (k_1, \dots, k_d) \in \mathbb{Z}_+^d$ mit $\sum k_i = n$ summiert.

Man kann dann mit etwas mehr Aufwand und der Stirling'schen Formel allgemein zeigen, dass $P^x(X_{2n} = x) \leq c \left(\frac{1}{n}\right)^{\frac{d}{2}}$ für ein $c < \infty$ gilt [GEO]. Da aber auch $\sum_{n \geq 0} \left(\frac{1}{n}\right)^{\frac{d}{2}} < \infty$ für $d \geq 0$ gilt, folgt dann schließlich

$$G(x, x) = \sum_{n \geq 0} P^x(X_{2n} = x) < \sum_{n \geq 0} \left(\frac{1}{n}\right)^{\frac{d}{2}} < \infty .$$

Damit sind nach Lemma (3.9) für $d \geq 3$ die Zustände in \mathbb{Z}^d nicht mehr rekurrent. Bereits bei einer symmetrischen Irrfahrt auf einem dreidimensionalen Gitter wird man also niemals wieder an den Ausgangspunkt zurückkehren.

Simulation der symmetrischen Irrfahrt auf \mathbb{Z}^2

Die Simulation einer Markov-Kette zu Beginn des Abschnitts soll nun für die Simulation einer symmetrischen Irrfahrt angewendet werden. Die folgende Funktion `RandomWalk` ist zwar allgemein für symmetrische Irrfahrten auf \mathbb{Z}^d definiert, jedoch ist nur im zweidimensionalen Fall eine graphische Darstellung implementiert.

Allerdings muß die Anwendung von `MarkovSimulation` leicht abgewandelt werden, denn da \mathbb{Z}^d nicht endlich ist, kann für eine entsprechende Markov-Kette keine Übergangsmatrix angegeben werden. Das Problem kann aber gelöst werden, indem die Markov-Kette nicht auf dem eigentlichen Zustandsraum \mathbb{Z}^d , sondern auf der Menge der Bewegungsrichtungen definiert wird. Für eine symmetrische Irrfahrt auf \mathbb{Z}^d gibt es nach Definition stets $2d$ Möglichkeiten, von einem Ausgangspunkt in einen nächsten zu wechseln und da diese Möglichkeiten auch als gleich wahrscheinlich definiert wurden, kann man also nun eine $(d \times d)$ -Matrix, welche in jedem Eintrag die Wahrscheinlichkeit $\frac{1}{2d}$ enthält für die Funktion `MarkovSimulation` angeben.

Um in der Graphik den Start- und den Endpunkt des Random Walks kenntlich zu machen, beginnt die Irrfahrt mit einem kleinen Aufstrich und endet mit einem kleinen Abstrich.

$$\Pi = (p_{ij})_{ij} = \begin{cases} p \frac{x}{N} & \text{falls } y = x - 1, \\ 1 - p & \text{falls } y = x, \\ p(1 - \frac{x}{N}) & \text{falls } y = x + 1, \\ 0 & \text{sonst.} \end{cases}$$

Im Folgenden soll allerdings nur der Fall $p = 1$ betrachtet werden, d.h. wurde ein Teilchen zufällig ausgewählt, so wechselt es auch in den anderen Behälter. Die Übergangswahrscheinlichkeit erhält somit die Form:

$$\Pi = \begin{pmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ \frac{1}{N} & 0 & \frac{N-1}{N} & & & 0 \\ 0 & \frac{2}{N} & 0 & \frac{N-2}{N} & & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & & & \frac{N-1}{N} & 0 \\ 0 & \dots & & & 0 & 1 \end{pmatrix}$$

Da man das Verhalten der Markov-Kette auf lange Sicht untersuchen möchte, wird nach einer stationären Verteilung der Markov-Kette gesucht. Um den Ergodensatz 3.6 anwenden zu können, muß dessen Voraussetzung erfüllt werden. Mit $m = N$ lässt sich aber bereits die geforderte Bedingung $\Pi^m = (p_{ij})_{ij}^m > 0$ zeigen. $\Pi_{(x,y)}$ beschreibt dabei den Eintrag der Matrix Π an der Stelle der Zustände x und y .

$$\Pi^N(x, y) \geq P^x(X_{N-|x-y|} = x) \cdot P^x(X_1 = x+1) \cdot P^{x+1}(X_1 = x+2) \cdot \dots \cdot P^{y-1}(X_1 = y) > 0$$

Die obige Ungleichung gilt für alle $x, y \in E$ mit $x \geq y$, für den Fall $x \leq y$ kann eine analoge Ungleichung gezeigt werden. Der erste Schritt der Ungleichung gilt, da auf der linken Seite nur ein Summand aus der Chapman-Kolmogorov-Gleichung 3.3 verwendet wird. Die zweite Abschätzung ist legitim, da die Faktoren gemäß der obigen Übergangsmatrizendefinition alle positiv sind.

Der Ergodensatz liefert nun also, dass der Limes $\lim_{n \rightarrow \infty} \Pi^n = a$ existiert und dessen Zeilen enthalten die einzige stationäre Verteilung der Markov-Kette. Somit ist die Existenz einer stationären Verteilung gesichert.

Bleibt noch zu zeigen, welche Form diese Verteilung annimmt [ZIE]:

Da für eine stationäre Verteilung a einer Markov-Kette $a = a\Pi$ gilt, und die Übergangsmatrix Π nicht sehr dicht besetzt ist, kann man einfach das lineare Gleichungssystem $a = a\Pi$ für a auflösen. Das Gleichungssystem sei allgemein definiert durch:

$$\begin{aligned} a_0 &= a p_{00} + a_1 p_{10} \\ a_1 &= a_0 p_{01} + a_1 p_{11} + a_2 p_{21} \\ a_2 &= a_1 p_{12} + a_2 p_{22} + a_3 p_{32} \\ &\dots \\ a_N &= a_{N-1} p_{N-1 N} + a_N p_{NN} \end{aligned}$$

Nach der Matrixdefinition von Π kann $p_{ij} > 0$ für $|x - y| = 1$ angenommen werden. Bei der Gleichung von a_1 beginnend werden nun die einzelnen Darstellungen der a_i so

umgeformt, dass sukzessiv die Gleichungen jeweils nur noch durch Matrixeinträge und a_0 dargestellt werden. Dabei wird die Eigenschaft einer stochastischen Matrix $p_{ij} = 1 - p_{ij}$ ausgenutzt.

$$\begin{aligned}
 a_1 &= \frac{1}{p_{10}}(a_0 - a_0 p_{00}) = a_0 \frac{p_{01}}{p_{10}}, & a_0 \frac{p_{01}}{p_{10}} &= a_0 p_{01} + a_0 \frac{p_{01}}{p_{10}} p_{11} + a_2 p_{21} \quad \Rightarrow \\
 a_2 &= a_0 \frac{p_{01}}{p_{10}} \frac{1 - p_{10} - p_{11}}{p_{10}} = a_0 \frac{p_{01} p_{12}}{p_{21} p_{10}}, & a_0 \frac{p_{01} p_{12}}{p_{21} p_{10}} &= a_0 \frac{p_{01}}{p_{10}} p_{12} + a_0 \frac{p_{01} p_{12}}{p_{21} p_{10}} p_{22} + a_3 p_{32} \quad \Rightarrow \\
 a_3 &= a_0 \frac{p_{01} p_{12} p_{23}}{p_{32} p_{21} p_{10}}, \quad \Rightarrow \\
 &\dots \\
 a_N &= a_0 \frac{p_{01} \cdot \dots \cdot p_{N-1 N}}{p_{N N-1} \cdot \dots \cdot p_{01}}
 \end{aligned}$$

Das Gleichungssystem wurde also so vereinfacht, dass es nur noch von a_0 abhängt. Zur Bestimmung von a_0 wird nun noch die Eigenschaft $\sum_{i=0}^N a_i = 1$ der Verteilung a bemüht. Damit kann in einer Gleichung mit einer Unbekannten eine eindeutige Lösung für a_0 gefunden werden und anschließend können durch Resubstitution sukzessiv die übrigen a_i berechnet werden.

Um Genaueres über das Aussehen der a_i zu erfahren wird die allgemeine Form

$$a_i = a_0 \frac{p_{01} \cdot \dots \cdot p_{i-1 i}}{p_{i i-1} \cdot \dots \cdot p_{01}}$$

betrachtet und die konkreten Matrixwerte von Π eingetragen. Es ergibt sich:

$$a_i = a_0 \frac{p(1 - \frac{1}{N}) \cdot p(1 - \frac{2}{N}) \cdot \dots \cdot p(1 - \frac{i}{N})}{p^{\frac{1}{N}} \cdot p^{\frac{2}{N}} \cdot \dots \cdot p^{\frac{i}{N}}} = a_0 \frac{(1 - \frac{1}{N}) \cdot (1 - \frac{2}{N}) \cdot \dots \cdot (1 - \frac{i}{N})}{\frac{1}{N} \cdot \frac{2}{N} \cdot \dots \cdot \frac{i}{N}} = a_0 \binom{N}{i}$$

Nun muß nur noch a_0 festgelegt werden:

$$\begin{aligned}
 1 &= \sum_{i=0}^N a_i = \sum_{i=0}^N a_0 \binom{N}{i} = a_0 (1 + 1)^N = a_0 2^N \\
 &\Rightarrow a_0 = \frac{1}{2^N}
 \end{aligned}$$

Die gesuchte stationäre Verteilung a hat letztlich die folgende Gestalt:

$$a = \left(\binom{N}{0} \frac{1}{2^N}, \binom{N}{2} \frac{1}{2^N}, \dots, \binom{N}{N} \frac{1}{2^N} \right)$$

Diese Form ist natürlich bekannt und entspricht der Binomial-Verteilung auf $E = \{0, \dots, N\}$, $N \in \mathbb{N}$.

Dieses Ergebnis bedeutet mit anderen Worten, dass nach sehr vielen Zeitschritten mit Wahrscheinlichkeit $\frac{1}{2}$ jedes Molekül sich in der Kammer A aufhält.

Um die genaue Gestalt einer stationären Verteilung einer konkreten Markov-Kette zu finden, kann natürlich Mathematica verwendet werden. Man berechnet dann eleganterweise nur mit dem Befehl `Eigensystem` alle Eigenvektoren und Eigenwerte der Matrix Π und wählt aus dessen Ausgabe den Eigenvektor zum Eigenwert 1. Dieser Vektor ist dann die stationäre Verteilung der Markov-Kette.

3.2.4 Markov-Ketten in der Genetik

Als weitere Anwendung in der Biologie finden Markov-Ketten auch in der Genetik Verwendung. Der konkrete Modellierungsaspekt ist dabei die Auswahl des Genmaterials bei der Kreuzung zweier Eltern-Zellen.

Das genetische Material besteht bei höheren Lebewesen im Normalfall aus einem diploiden Chromosomensatz, dies bedeutet, dass die genetische Information in zweifacher Ausführung auf so genannten Chromosomen im Zellkern gespeichert wird. Da beide Elternteile je einen Gensatz beisteuern, ist jedes Gen paarweise vorhanden. An einem Genort auf einem Chromosomenarm kann ein Gen in verschiedenen Ausprägungen vorliegen. Diese unterschiedlichen Zustandsformen eines Gens werden als Allele bezeichnet. Da also für jedes Gen zwei Allele, eines vom Vater, eines von der Mutter, vorliegen, muß bei der Realisierung des Gens entweder eine Mischform beider Allele, oder, wie in den allermeisten Fällen, eines von beiden für die Ausprägung ausgewählt werden. Als Genotyp werden die beiden vorhandenen Allele eines Gens bezeichnet, der Phänotyp eines Gens ist das für die Realisierung ausgewählte Allel.

Bei der Befruchtung einer Eizelle mischt sich das Genmaterial der Eltern. Auf zufällige Weise werden aus den beiden diploiden Chromosomensätzen jeweils vom Vater und der Mutter ein vollständiger Gensatz ausgewählt und an die Tochterzelle als deren Genotyp vererbt. Diese Auswahl findet völlig zufällig statt, daher ist jedes Ergebnis eines Genotyps gleich wahrscheinlich. Beschränkt man sich auf die Betrachtung eines Gens, und besitzen die Eltern je unterschiedliche Allele, so gibt es für den Genotyp der Tochterzelle vier verschiedene Möglichkeiten.

Eine Anwendung von Markov-Ketten gibt es nun in der Pflanzenzüchtung. Angenommen, ein Pflanzengemisch besitzt die beiden Allele A und a und man möchte durch Zuchttechnik möglichst reinrassige, homozygote Ergebnisse, d.h. Pflanzen mit zwei gleichen Allelen erzeugen. In der Botanik und im Gartenbau ist die Methode der Selbstbefruchtung, welche Pflanzen mit sich selbst kreuzt ist dafür ein bewährtes Verfahren. Mit den Eigenschaften einer Markov-Kette kann man zeigen warum.

Sei $(X_n)_{n \geq 0}$ eine Markov-Kette mit Zustandsraum $E = \{AA, Aa, aa\}$ und der Übergangsmatrix

$$\Pi = \begin{pmatrix} 1 & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & 1 \end{pmatrix}.$$

Dies bedeutet, ist die Zuchtpflanze bereits homozygot, so bleibt sie es auch, denn die Zustände AA und aa sind absorbierende Zustände. Dagegen bei heterozygoten Genotypen ist die Wahrscheinlichkeit heterozygot zu bleiben $\frac{1}{2}$, und für einen Wechsel zu einem der beiden homozygoten Fälle AA oder aa beträgt sie je $\frac{1}{4}$.

Mit Hilfe von Mathematica soll nun die Wahrscheinlichkeit

$$P^{Aa}(X_n = Aa) = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \Pi^n(Aa)$$

berechnet werden.

$$\text{In}[1]:= \mathbf{ma} = \begin{pmatrix} 1 & 0 & 0 \\ 1/4 & 1/2 & 1/4 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\text{Out}[1]= \left\{ \{1, 0, 0\}, \left\{ \frac{1}{4}, \frac{1}{2}, \frac{1}{4} \right\}, \{0, 0, 1\} \right\}$$

$$\text{In}[2]:= \mathbf{mp} = \text{MatrixPower}[\mathbf{ma}, n]$$

$$\text{Out}[2]= \left\{ \{1, 0, 0\}, \left\{ \frac{1}{2} - 2^{-1-n}, 2^{-n}, \frac{1}{2} - 2^{-1-n} \right\}, \{0, 0, 1\} \right\}$$

$$\text{In}[3]:= \{0, 1, 0\} \cdot \mathbf{mp}$$

$$\text{Out}[3]= \left\{ \frac{1}{2} - 2^{-1-n}, 2^{-n}, \frac{1}{2} - 2^{-1-n} \right\}$$

$$\text{In}[4]:= \mathbf{l} = \{0, 1, 0\} \cdot \mathbf{mp}[[2]]$$

$$\text{Out}[4]= 2^{-n}$$

$$\text{In}[5]:= \text{Limit}[\mathbf{l}, n \rightarrow \infty]$$

$$\text{Out}[5]= 0$$

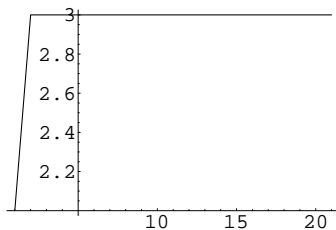
Die Wahrscheinlichkeit bei einem Start mit einer Pflanze des Genotyps *Aa* nach sukzessiver Selbstkreuzung wieder einen heterozygoten Genotyp zu erhalten, geht also für $n \rightarrow \infty$ exponentiell gegen Null. Selbst bei einem Start mit gemischten Allelen scheint die Methode der Selbstbefruchtung also bereits nach wenigen Iterationen eine praktikable Züchtungsart für reinrassigen Pflanzenarten zu sein.

Auch die Markov-Ketten-Simulation zeigt dieses Verhalten:

MarkovPlot[[{1, 0, 0}, {0.25, .5, .25}, {0, 0, 1}], 2, 20, 1, 1]

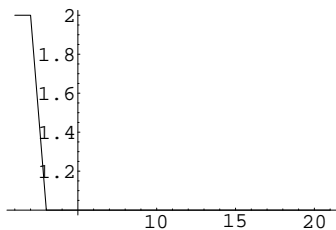
$$\begin{pmatrix} 1 & 0 & 0 \\ 0.25 & 0.5 & 0.25 \\ 0 & 0 & 1 \end{pmatrix}$$

{2, 3}



MarkovPlot[[{1, 0, 0}, {0.25, .5, .25}, {0, 0, 1}], 2, 20, 0, 1]

{2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1}



Verwendete Literatur

Die grundlegenden Literaturquellen dieses letzten Kapitels waren für die theoretischen Einführungen hauptsächlich [GEO] und [ZIE]. Außerdem fand im Abschnitt der Anwendungen neben diesen beiden auch [MUE] Verwendung.

Kapitel 4

Fazit

Ich hoffe mit dieser Arbeit ist es mir gelungen, verschiedene Aspekte mathematischer Modellierung in der Biologie vorzustellen. Ich habe dabei versucht unterschiedliche Themenbereiche, interessante Problematiken und theoretische Modellierungsansätze auszuwählen, da das Gebiet mathematischer Modellierung in der Biologie natürlich ein weites Feld ist.

Während des Erstellens der Arbeit zeigte sich jedoch an vielen Stellen, dass geplante Vorhaben aufgrund ihrer zu großen Komplexität den Rahmen der Arbeit sprengen würden und viel theoretische Vorarbeit nötig wäre, um die Themen vollständig zu durchleuchten. Deshalb wurden in der Arbeit oft elementare Grundprinzipien von Modellierungsansätzen vorgestellt, mit dem Verweis darauf, in welcher Richtung eine intensivere Behandlung des Problems weiter möglich wäre.

Bei allen Ansätzen erwies sich in jedem Fall *Mathematica* als ein sehr vorteilhaftes Hilfsmittel. Schnelle Berechnungen, Visualisierungen und auch bei der vorgenommenen Simulation konnten oft Prognosen bestärkt und neue Erkenntnisse gewonnen werden.

Literaturverzeichnis

- [AUL] B. AULBACH, *Gewöhnliche Differentialgleichungen*
Spektrum Akademischer Verlag, Heidelberg - Berlin, 1997.
- [BRA] M. BRAUN, *Differential equations and their applications*
Springer-Verlag, New York, 1983, 3. Auflage.
- [GEO] H.-O. GEORGI, *Stochastik*
De Gruyter, Berlin, 2004, 2. Auflage.
- [GRU] L. GRUENE, *Vorlesungsskript Modellierung in der Biologie*
<http://www.uni-bayreuth.de/departments/math/lgruene/modellierung03/>
Stand 11.7.2003
Universität Bayreuth, SS 2003.
- [HEU] H. HEUSER, *Gewöhnliche Differentialgleichungen*
B. G. Teubner, Stuttgart, 1995, 3. Auflage.
- [HOP] F. C. HOPPENSTEADT, *An introduction to the mathematics of neurons*
Cambridge University Press, Cambridge, 1986.
- [HOP/PES] F. C. HOPPENSTEADT, C. S. PESKIN *Mathematics in medicine and the life sciences*
Springer-Verlag, New York, 1996, 3. Auflage.
- [KOE] W. KOEPF, *Vorlesungsskript Computeralgebra II: Kapitel Modellierung in der Biologie*
Universität Kassel, SS 2003.
- [MUE] J. MÜLLER, *Vorlesungsskript Mathematical Models in Biology*
<http://www-m12.ma.tum.de/lehre/model2003/skript.php>
Stand: 22.01.2004, 15:04
Technische Universität München, WS 2003/2004.
- [MUR] J. D. MURRAY, *Mathematical Biology, I: An Introduction*
Springer-Verlag, Heidelberg - Berlin, 2002, 3. Auflage.
- [YSH] E. K. YEARGERS, R. W. SHONKWILER, J. V. HEROD, *An introduction to the mathematics of biology with computer algebra models*
Birkhäuser, Boston, 1996.

[WWW] <http://de.wikipedia.org/wiki/Bild:Aktionspotenzial.png>
Auf dieser Seite seit 3.5.2005.

[ZIE] H. ZIEZOLD, *Vorlesungsmitschrift Stochastik III*
Universität Kassel, WS 2004/2005.

Ich erkläre hiermit, dass ich die vorliegende Arbeit ohne fremde Hilfe und nur unter der Verwendung der im Literaturverzeichnis angegebenen Quellen und Hilfsmitteln erstellt habe.

Kassel, den 1. September 2005