

Dynamische Systeme

Die Mathematik zum Computerprogramm «Simulator»

Das Computerprogramm «Simulator», dessen Einsatz in einem separaten Handbuch beschrieben wird, ermöglicht die Simulation einfacher dynamischer Systeme und das Experimentieren mit ihnen. Der Code ist öffentlich auf GitHub zugänglich, in VB.NET geschrieben, mit ausführlichen Kommentaren versehen und kann nach Bedarf erweitert werden. Dazu ist die kostenlose Community-Version von Microsoft Visual Studio nötig. Dieses Dokument beschreibt die mathematischen Grundlagen für den «Simulator» auf elementarem Niveau.

Version 4.0 - 01.04.2024

Inhalt

Einführung	3
1. Historischer Hintergrund	4
2. Dynamische Systeme	5
2.1. Grundlagen.....	5
2.2. Orbit und periodische Punkte	6
2.3. Attraktoren und Repelloren.....	7
2.4. Chaotische dynamische Systeme	8
2.5. Histogramme.....	9
2.6. Protokolle.....	10
3. Beispiele diskreter dynamischer Systeme	12
3.1. Das Bernoulli-Shift-System.....	12
3.2. Die Zeltabbildung	13
3.3. Das logistische Wachstum	20
3.4. Konjugierte der Zeltabbildung	23
3.5. Die «normierte» Parabel.....	27
3.6. Implementierung im «Simulator».....	29
3.7. Histogramme im chaotischen Fall.....	31
3.8. Zweidimensionale Darstellung und Transitivität im chaotischen Fall	32
3.9. Unimodale Funktionen	33
3.10. Periodenverdoppelung	34
3.11. Implementierung im «Simulator».....	40
3.12. Übungsbeispiele.....	41
4. Mathematisches Billard	43
4.1. Einführung.....	43
4.2. Grundlagen des elliptischen Billards.....	44

4.3.	Implementierung des elliptischen Billards im «Simulator»	50
4.4.	Billard im Kreis	51
4.5.	Periodische Punkte beim Ellipsenbillard.....	55
4.6.	Kaustik im elliptischen Billard	59
4.7.	Billard im Stadion.....	62
4.8.	Implementierung im «Simulator».....	67
4.9.	Ovales Billard	67
4.10.	Darstellung des Phasenraumes.....	71
4.11.	Der konvexe Billardtisch	72
4.12.	Das C-Diagramm	76
4.13.	Übungsbeispiele.....	83
5.	Numerische Methoden zur Lösung von gewöhnlichen Differentialgleichungen	86
5.1.	Gewöhnliche Differentialgleichungen	86
5.2.	Das klassische Federpendel	87
5.3.	Das explizite Euler Verfahren.....	88
5.4.	Das implizite Euler Verfahren	91
5.5.	Die implizite Mittelpunktregel	92
5.6.	Das Runge-Kutta Verfahren vierter Ordnung	94
5.7.	Implementierung im «Simulator».....	96
5.8.	Übungsbeispiele.....	98
6.	Gekoppelte Pendel	100
6.1.	Der Lagrange-Formalismus	100
6.2.	Das Doppelpendel.....	105
6.3.	Implementierung des Doppelpendels.....	108
6.4.	Untersuchung des Phasenraumes für das Doppelpendel.....	112
6.5.	Schwingendes Federpendel	115
6.6.	Implementierung des schwingenden Federpendels.....	117
6.7.	Rüttelpendel	121
6.8.	Übungsbeispiele.....	123
7.	Iteration in der komplexen Ebene	125
7.1.	Einführung.....	125
7.2.	Das Newton Verfahren	125
8.	Implementierung eigener Systeme im «Simulator».....	130
	Weiterführende Literatur.....	132

Einführung

Im Internet findet man viele Programme, welche die Simulation von einfachen dynamischen Systemen ermöglichen. Deren Code ist aber kaum öffentlich und die zugrundeliegende Mathematik ist auch wenig dokumentiert. Der «Simulator» ermöglicht in der vierten Version die Iteration von unimodalen Funktionen, die Simulation des mathematischen Billards, die Untersuchung von numerischen Verfahren zur Lösung von gewöhnlichen Differentialgleichungen und die Simulation von verschiedenen gekoppelten Pendeln. Der Code des Programmes ist in VB.NET geschrieben und steht öffentlich auf Github im Repository «HermannBiner/Simulator» zur Verfügung. Um mit ihm zu arbeiten, genügt die Community Version 2022 des Microsoft Visual Studio, welche kostenlos heruntergeladen und einfach installiert werden kann.

Die Mathematik, welche dem «Simulator» zu Grunde liegt, wird in diesem Dokument auf elementarem Niveau behandelt. Beispiele von Übungsaufgaben oder Vorschläge zur Erweiterung des «Simulator» sollen zu eigener Aktivität anregen. Die Mathematik am Gymnasium ist stofflich zwar mehr als ausgelastet. Dennoch kann es sein, dass im Rahmen eines Freifaches oder eines Seminars interessierten Schülern weiterführende Themen angeboten werden. Der «Simulator» und dieses Dokument soll einen Beitrag dazu bieten.

In der Version 4.0 des Programmes sind folgende Systeme implementiert:

- Wachstumsmodelle und Iteration unimodaler Funktionen wie das logistische Wachstum, die Zeltabbildung, die Iteration an der Parabel inklusive der damit verbundenen Themen wie das Feigenbaum Diagramm.
- Das mathematische Billard mit verschiedenen Formen des Billardtisches: Elliptisches Billard, Billard im Stadion, ovales Billard. Das Analogon zum Feigenbaum-Diagramm ist hier das C-Diagramm.
- Die Untersuchung von numerischen Verfahren zur Lösung von gewöhnlichen Differentialgleichungen. Hier werden einige einfache Verfahren am Beispiel des Federpendels miteinander verglichen.
- Die Simulation von gekoppelten Pendeln: Doppelpendel, schwingendes Federpendel und horizontales Rüttelpendel.

Der Einsatz des «Simulator» ist in einem Handbuch auf Deutsch und einem Manual auf Englisch ausführlich dokumentiert. Die Benutzersprache kann zwischen Deutsch und Englisch gewählt werden. Im Code findet man ausführliche Kommentare auf Englisch. Dieses Dokument mit den mathematischen Kommentaren ist auf Deutsch und Englisch verfügbar.

Vorausgesetzt wird lediglich Mathematik, welche entweder am Gymnasium behandelt wird oder welche einem Mittelschüler mit wenig Aufwand zugänglich gemacht werden kann. Gestreift werden Themen aus der Geometrie (Kegelschnitte und ebene Vektorgeometrie), der Analysis (Stetigkeit, Differentialrechnung und gewöhnliche Differentialgleichungen) und der Physik (Lagrange Formalismus).

Die einzelnen Themenbereiche sind weitgehend unabhängig, sodass je nach Interesse und der zur Verfügung stehenden Zeit eine Auswahl möglich ist. Abgesehen vom ersten einführenden Kapitel 2 können die folgenden isoliert ausgewählt werden.

Der «Simulator» soll in kommenden Versionen thematisch ausgebaut und mit weiteren dynamischen Systemen erweitert werden.

1. Historischer Hintergrund

Die klassische Mechanik als Lehre bewegter Körper begann 1687, als Isaac Newton seine «Principia» veröffentlichte. 1736 formulierte Leonhard Euler Newtons Mechanik mathematisch präziser und vereinigte sie mit der Infinitesimalrechnung von Leibniz. 1788 führte Joseph-Louis Lagrange im Bereich der klassischen Mechanik die Lagrange Funktion ein, welche auf dem Prinzip der kleinsten Wirkung beruht und mit welcher Bewegungsgleichungen beliebiger mechanischer Systeme aufgestellt werden können. Pierre-Simon Laplace brachte 1796 die Himmelsmechanik auf einen Stand, der es erlaubte, die bis anhin unerklärten Bahnstörungen des Uranus auf den bis damals unentdeckten Planeten Neptun zurückzuführen.

Beeindruckt von diesen grossen Fortschritten der Mechanik während rund hundert Jahren formulierte Laplace 1814 eine Aussage, welche sinngemäss sagt, dass ein mit genügend grosser Intelligenz ausgestatteter Dämon auf Grund der Gesetze der Mechanik und mit genügend genauer Kenntnis des jetzigen Zustandes der Welt die gesamte Zukunft derselben voraussagen könne. Zukunft und Vergangenheit würden klar vor Augen dieser Intelligenz liegen. Diese hypothetische Intelligenz wurde auch als «Laplace'scher Dämon» bekannt.

Zweifel an dieser Auffassung kamen später an verschiedenen Stellen auf:

- Beim Studium des Dreikörperproblems (wie bewegen sich drei Himmelskörper um ihren gemeinsamen Schwerpunkt?) vermutete Henri Poincaré um 1888, dass dieses System so sensibel auf die Anfangsbedingungen reagiert, dass sein Verhalten nur mit immer grösserem Aufwand längerfristig vorausgesagt werden kann. Eine geschlossene mathematische Lösung (wie etwa die Ellipsenbahnen beim Zweikörperproblem) sei nicht möglich.
- Gemäss der Relativitätstheorie (1905) von Albert Einstein ist es nicht möglich, den ganzen Kosmos zu erfassen. Auf Grund der endlichen Lichtgeschwindigkeit bildet sich ein Erkenntnisthorizont, über den der «Dämon» nicht hinausblicken könnte.
- Werner Heisenberg formulierte in einer 1927 veröffentlichten Arbeit die sogenannte Unschärferelation. Sie sagt im Wesentlichen, dass auf Quantenebene keine deterministischen Aussagen möglich sind, sondern lediglich Wahrscheinlichkeitsaussagen.
- Die Chaostheorie, welche Anfang des letzten Jahrhunderts mit Henri Poincaré und seinem Studium des Dreikörperproblems ihren Anfang nahm, entdeckte ebenfalls Phänomene, welche den «Dämon» vor unlösbare Aufgaben stellen. In den 1960-er Jahren bemerkte der Mathematiker und Meteorologe Edward Lorenz bei der Computersimulation des Wettergeschehens, dass minimale Störungen des Systems dramatische Auswirkungen auf dessen zukünftige Entwicklung haben können. Er prägte 1972 das Bild vom Schmetterling, dessen Flügelschlag einen Orkan auslösen kann.
- Seit etwa Mitte des letzten Jahrhunderts sind deshalb dynamische Systeme in den Fokus der mathematischen Forschung gerückt. Das interessante dabei ist, dass sich gewisse Phänomene bereits auf Mittelschulniveau behandeln lassen. Dazu erschienen in den 1990-er Jahren mehrere Beiträge aus dem Programm «ETH für die Schule». Siehe Literaturverzeichnis.

2. Dynamische Systeme

2.1. Grundlagen

Bevor wir uns den im «Simulator» implementierten Beispielen zuwenden, ist es nötig, einige Grundbegriffe einzuführen.

Dynamische Systeme sind Systeme (oft physikalischer Art), deren Zustand sich im Laufe der Zeit ändert, die aber einem *Bewegungsgesetz* gehorchen. Beispiele sind etwa die Planetenbewegung, das Doppelpendel, aber auch Modelle des Wettergeschehens. Diese genannten Beispiele sind *kontinuierliche dynamische Systeme*.

Es kann sein, dass die Zustandsänderung durch die *Iteration einer Abbildung* erfolgt. Dann spricht man von einem *diskreten dynamischen System*. Ein Beispiel ist das bekannte *3n+1 Problem (auch Collatz Problem genannt)*.

Beim 3n+1 Problem betrachtet man die Iteration der «3n+1 Abbildung»:

$$f(n) = \begin{cases} \frac{n}{2}, & n \text{ gerade} \\ \frac{3n + 1}{2}, & n \text{ ungerade} \end{cases}, n \in \mathbb{N}$$

Die offene Frage ist, ob sich die Folge unabhängig vom Startpunkt der Zahl 1 und damit dem Zyklus 1, - 2, -1, -2, nähert.

Allgemein kann man definieren:

Ein *dynamisches System* ist ein Tripel (T, X, f) . Die Menge T ist der *Zeitraum*. Typischerweise ist $T = \mathbb{N}_0$ oder \mathbb{Z} bei diskreten dynamischen Systemen und $T = \mathbb{R}_0^+$ oder \mathbb{R} bei kontinuierlichen dynamischen Systemen. Die Menge X ist der *Zustandsraum* und f eine Abbildung: $f: X \rightarrow X$.

Bei *diskreten dynamischen Systemen* geht ein Zustand im n-ten Schritt über in:

$$x_{n+1} = f(x_n), x_n \in X, n \in T = \mathbb{N}_0 \text{ oder } \mathbb{Z}$$

Meistens ist X ein reelles Intervall oder eine Teilmenge des \mathbb{R}^n .

Bei *kontinuierlichen dynamischen Systemen* hängt der «nächste» Zustand des Systems vom aktuellen Zustand und von der «momentanen Tendenz» an dieser Stelle ab. Letztere wird in der Regel durch ein System von Differentialgleichungen beschrieben. Diese sind dann interessant, wenn sie nicht integrierbar sind. Man versucht dann, das Verhalten des Systems schrittweise durch numerische Verfahren zu approximieren. Durch diese «stroboskopische Betrachtung» wird ein kontinuierliches dynamisches System in ein entsprechendes diskretes dynamisches System überführt.

Wir werden uns hier auf diskrete dynamische Systeme beschränken.

Die Grundfrage bei diesen Untersuchungen ist die *Prognose der Systemzukunft*.

Idee: Wenn das System durch ein Bewegungsgesetz präzis beschrieben wird, hängt die globale Systemzukunft nur noch von den Anfangsbedingungen ab. Das ist der Determinismus im strengen Sinne des Laplace'schen Dämon.

Da man die Anfangsbedingungen aber nicht unendlich genau messen kann, hofft man darauf, dass sich diese Ungenauigkeit nicht gross auswirkt. Je nachdem hat man ein:

«Braves System»: Es gilt das starke Prinzip von Ursache und Wirkung: *Ähnliche Ursachen haben ähnliche Wirkungen.*

«Chaotisches System»: Es gilt das schwache Prinzip von Ursache und Wirkung: *Nur exakt gleiche Ursachen haben gleiche Wirkungen.* Ähnliche Ursachen können dramatische unterschiedliche Wirkungen haben!

Eine umfangreiche Einführung in das Verhalten dynamischer Systeme findet man in [1]. [2] bietet einen guten Zugang auf Mittelschulniveau und enthält Übungsbeispiele. Wir wollen im Folgenden die wichtigsten Begriffe rekapitulieren.

2.2. Orbit und periodische Punkte

Wir betrachten ein dynamisches System (T, X, f) . Wenn wir einen Startpunkt $x_0 \in X$ wählen, entsteht im Laufe der Bewegung eine Bahn dieses Punktes. Diese bezeichnet man als *Orbit* von x_0 . Bei einem diskreten dynamischen System ist dies:

$$Or^+(x_0) := \{f^n(x_0) \in X, n \in \mathbb{N}_0\}$$

Umgekehrt kann man die Menge aller Punkte betrachten, welche im Laufe der Iteration auf x_0 fallen. Diese Menge heisst *inverser Orbit* von x_0 . Bei einem diskreten dynamischen System ist dies:

$$Or^-(x_0) := \{x \in X / \exists n \in \mathbb{N} \text{ mit } f^n(x) = x_0\}$$

Beispiel

Bei der «3n+1 Abbildung» ist:

$$Or^+(7) = \{7, 11, 17, 26, 13, 20, 10, 5, 8, 4, 2, 1\}$$

Eine offene Frage ist: $Or^-(1) = \mathbb{N} ?$

Es kann sein, dass für einen Punkt ξ gilt: $f(\xi) = \xi$. Dann heisst ξ *Fixpunkt* der Iteration. Analog dazu heisst ξ ein *periodischer Punkt der Ordnung k oder k-periodischer Punkt von f*, falls dieser Punkt nach minimal k Iterationsschritten wieder auf sich abgebildet wird. Für einen derartigen Punkt gilt:

$$f^k(\xi) = \xi, k \in \mathbb{N} \text{ und } f^i(\xi) \neq \xi \text{ für } i \in \mathbb{N} \text{ und } 0 < i < k$$

Ein k-periodischer Punkt von f ist ein Fixpunkt von f^k .

Wenn ξ ein k-periodischer Punkt von f ist, dann heisst die Menge

$$\{\xi, f(\xi), f^2(\xi), \dots, f^{k-1}(\xi)\} =: \{\xi_1, \xi_2, \dots, \xi_k\}$$

k-Zyklus der Iteration von f . Jeder Punkt in diesem Zyklus ist dann ein k-periodischer Punkt:

$$f^k(\xi_i) = f^k(f^{i-1}(\xi_1)) = f^{k+i-1}(\xi_1) = f^{i-1}(f^k(\xi_1)) = f^{i-1}(\xi_1) = \xi_i, \text{ für } i \in \mathbb{N} \text{ und } 1 \leq i \leq k$$

Beispiel

Bei der «3n+1 Abbildung» ist: {2, 1} ein 2-Zyklus.

2.3. Attraktoren und Repelloren

Interessant bei einem dynamischen System ist das *asymptotische* Verhalten der Iteration ausgehend von einem Startpunkt $x_0 \in X$. Bei einem diskreten dynamischen System ist dies das Verhalten der Iteration, wenn die Anzahl der Iterationsschritte gegen unendlich strebt.

Es kann sein, dass für einen Fixpunkt $\xi \in X$ und einen Startwert $x_0 \in X$ gilt:

$$\lim_{n \rightarrow \infty} f^n(x_0) = \xi$$

Dann heisst x_0 *asymptotisch periodisch* (mit Periode 1) und analog:

x_0 heisst *asymptotisch periodisch mit Periode k*, falls der Orbit von x_0 gegen einen k-Zyklus konvergiert. Das heisst, dass der Orbit in konvergente Teilfolgen zerfällt, von denen jede gegen einen Punkt des Zyklus konvergiert. Oder auch, dass x_0 asymptotisch periodisch mit Periode 1 unter der Iteration f^k ist.

Entscheidend für das Verhalten der Iteration in der Nähe eines Fixpunktes (oder k-periodischen Punktes) ist der Wert der Ableitung von f in diesem Punkt, wenn f stetig differenzierbar ist.

Sei $I \subset \mathbb{R}$ ein reelles Intervall, ξ ein Fixpunkt von f und $f: I \rightarrow I$ stetig differenzierbar. Dann heisst ξ *attraktiver Fixpunkt* oder kurz *Attraktor*, falls gilt: $|f'(\xi)| < 1$. Der Wert $\lambda := f'(\xi)$ heisst *Multiplikator* des Fixpunktes ξ . Ist sogar $|f'(\xi)| = 0$, heisst ξ *superattraktiv*.

Wenn ein Punkt im Laufe der Iteration einem Attraktor ξ «zu nahe» kommt, dann kann er nicht mehr aus seiner Umgebung «entfliehen», sondern wird asymptotisch gegen diesen Attraktor streben.

Begründung: Sei ξ ein Attraktor. Aus Stetigkeitsgründen gilt dann für eine offene Umgebung

$$U(\xi) \subset I: |f'(x)| < 1 \text{ für } x \in U(\xi)$$

Wenn nun ein Punkt im Laufe der Iteration in diese Umgebung fällt, also für einen Startwert x_0 ein $n \in \mathbb{N}$ existiert mit $x_n = f^n(x_0) \in U(\xi)$, dann gilt nach dem Mittelwertsatz, dass es ein $\vartheta \in U(\xi)$ gibt mit

$$|x_{n+1} - \xi| = |f(x_n) - f(\xi)| = |f'(\vartheta)| |x_n - \xi| < L |x_n - \xi| \text{ für ein gewisses } L < 1$$

Bei der weiteren Iteration gilt dann:

$$|x_{n+k} - \xi| < L^k |x_n - \xi| \rightarrow 0 \text{ für } k \rightarrow \infty$$

Und die Folge $(x_{n+k})_k$ strebt gegen den Fixpunkt ξ .

In Zusammenhang mit einem Attraktor ξ spricht man vom *Bassin* des Attraktors. Dieses besteht aus allen Punkten, welche asymptotisch gegen den Attraktor konvergieren, also:

$$B(\xi) := \{x \in X / \lim_{n \rightarrow \infty} f^n(x) = \xi\}$$

Analog kann man vom Bassin eines attraktiven Zyklus sprechen.

Sei $I \subset \mathbb{R}$ ein reelles Intervall, ξ ein Fixpunkt von f und $f: I \rightarrow I$ stetig differenzierbar. Dann heisst ξ *repulsiver Fixpunkt* oder kurz *Repellor*, falls für dessen Multiplikator gilt: $|\lambda| = |f'(\xi)| > 1$.

Wenn ein Punkt im Laufe der Iteration einem Repellor ξ «nahe» kommt, dann wird er sich bei den nächsten Iterationsschritten wieder von ξ weg bewegen, mindestens bis er aus der Umgebung $U(\xi)$ mit $|f'(x)| > 1$ für $x \in U(\xi)$ «entflohen» ist. Vielleicht gerät er später wieder in diese

Umgebung, aber wird dann wieder hinauskatapultiert. Wenn er nicht genau auf den Fixpunkt ξ trifft, wird er ihn nie asymptotisch erreichen.

Falls ξ ein Fixpunkt ist mit Multiplikator $|\lambda| = |f'(\xi)| = 1$, dann heisst ξ *indifferent*.

Beispiel

Betrachte die Abbildung: $f(x) = 2x \bmod 1, [0,1] \rightarrow [0,1]$

Sie ist auch bekannt als *Bernoulli-Shift-System*.

Wenn man x als Dualbruch schreibt, dann sieht man die Wirkung der Abbildung. Sei $s_i \in \{0,1\}$ die i -te Ziffer von x in Dualbruchdarstellung. Dann gilt:

$$f: 0, s_1 s_2 s_3 \dots \rightarrow 0, s_2 s_3 \dots$$

0 ist ein Fixpunkt dieser Abbildung. Jeder periodische Dualbruch ist ein Zyklus mit seiner Periodenlänge. Jeder präperiodische Dualbruch ist ein präperiodischer Zyklus. Sämtliche Zyklen sind repulsiv, denn $f'(x) = 2 > 1$ auf $[0,1]$. Das Bernoulli-Shift-System wird später noch eingehender behandelt.

Bei einem Zyklus stellt sich die Frage, ob ein Zyklus sowohl aus Attraktoren wie auch Repelloren bestehen kann. Betrachten wir einen Zyklus

$$\{\xi, f(\xi), f^2(\xi), \dots, f^{k-1}(\xi)\}$$

Dann ist jeder Punkt des Zyklus ein Fixpunkt von f^k . Ob ein Punkt $\xi_i := f^i(\xi_1)$, $1 \leq i < k$ attraktiv ist oder nicht hängt von der Ableitung von f^k an dieser Stelle ab, da ξ_i ein Fixpunkt von f^k ist. Nach der Kettenregel gilt:

$$[f^k(\xi_i)]' = f' \left(f^{k-1}(\xi_i) \right) \cdot f' \left(f^{k-2}(\xi_i) \right) \cdots f'(\xi_i)$$

Die Funktionsargumente auf der rechten Seite durchlaufen aber gerade den ganzen Zyklus. Wenn man umordnet, gilt nämlich:

$$\{f^{k-1}(\xi_i), f^{k-2}(\xi_i), f^{k-3}(\xi_i), \dots, \xi_i\} = \{\xi_1, f(\xi_1), f^2(\xi_1), \dots, f^{k-1}(\xi_1)\} = \{\xi_1, \xi_2, \dots, \xi_k\}$$

Somit ist die Ableitung

$$[f^k(\xi_i)]' = f'(\xi_k) \cdot f'(\xi_{k-1}) \cdots f'(\xi_1) = \prod_{i=1}^k f'(\xi_i)$$

unabhängig von i und nimmt für alle Punkte des Zyklus denselben Wert an. Alle Punkte des Zyklus haben also *denselben* Multiplikator und sind alle entweder *attraktiv* oder *repulsiv*.

2.4. Chaotische dynamische Systeme

Die Beispiele, welche im «Simulator» implementiert sind, weisen teilweise sogenanntes chaotisches Verhalten auf. Deshalb wollen wir uns auch mit diesem beschäftigen. Weiterführende Erläuterungen sowie verschiedene Definitionen von chaotischen Systemen findet man in [1], [2] oder [3].

Wir wollen in diesem Dokument folgende Definition verwenden, welche auf Robert L. Devaney (1989) zurückgeht:

Ein dynamisches System heisst *chaotisch*, falls es folgende Eigenschaften erfüllt:

- 1) Es ist *sensitiv* abhängig von den Anfangsbedingungen. Das heisst, zwei beliebig nahe beieinander liegende Startwerte $x_1, x_2 \in X$ entfernen sich im Laufe der Iteration im Zustandsraum X beliebig weit auseinander.
- 2) Es ist *transitiv*. Das heisst, wenn ein beliebiger Startwert $x_0 \in X$ und ein beliebiger Zielwert $z \in X$ gegeben sind, dann gibt es einen alternativen Startwert in einer beliebig kleinen Umgebung von x_0 der im Laufe der Bewegung dem Zielwert z beliebig nahekommt.
- 3) Die periodischen Orbits liegen *dicht* im Zustandsraum.

Die *Sensitivität* ist das, was der Mathematiker und Meteorologe Edward Lorenz 1963 beobachtet hat. Er stellte ein einfaches Differentialgleichungssystem auf, welche das Wettergeschehen beschreiben sollte. Mit Computerhilfe wollte er daraus eine Wetterprognose erstellen. Da der Computer sehr viel Rechenzeit benötigte, rundete Lorenz die Anfangsbedingungen auf weniger Rechenstellen. Diese Rundungen führten zu einer komplett anderen Wettervoraussage. Er formulierte dann den sogenannten *Schmetterlingseffekt*, der sagt, dass *der Flügelschlag eines Schmetterlings in Brasilien einen Tornado in Texas auslösen kann*.

Die *Transitivität* hat dramatische Auswirkungen auf einen Experimentator. Messinstrumente haben eine gewisse Genauigkeit. Nehmen wir an, alle Werte innerhalb einer sehr kleinen Umgebung $U_\delta(x) \subset X$ werden bei der Messung als *derselbe* Messwert angezeigt. Dann gibt es für jeden «Zielpunkt» $z \in X$ einen Startpunkt $x_0 \in U_\delta(x)$ welcher im Laufe der Iteration in eine Umgebung $U_\varepsilon(z)$ hineinfällt und bei der Messung als z angezeigt wird, wenn $\varepsilon < \delta$. Bei diesem Experiment wird also jeder Startwert, der unterhalb der Messgenauigkeit des Experimentators liegt, als *derselbe* Startwert erscheinen. Er kann aber bei jedem beliebigen vorgegebenen Zielwert landen. Dieses System erscheint dem Experimentator völlig zufällig und eine Kausalität ist nicht erkennbar. Das Experiment kann bei jeder Wiederholung ein völlig unterschiedliches Resultat ergeben und das, obwohl dem dynamischen System eigentlich ein präzises Bewegungsgesetz zu Grunde liegt. Man spricht in diesem Zusammenhang auch von Pseudozufall oder deterministischem Chaos. Die periodischen Orbits sind alle repulsiv, sonst wäre die Sensitivität und Transitivität nicht für ganz X gegeben. Wenn diese dicht in X liegen, dann werden alle aperiodischen Punkte dauernd «wegkatapultiert», weil sie dauernd beliebig nahe an einen Repellor kommen.

Bei einem chaotischen System weist der Orbit eines Startpunktes unendlich viele Häufungspunkte auf und ist wie zufällig über den Zustandsraum verteilt. Man erhält einen sogenannten *seltsamen Attraktor*. Es gibt im aperiodischen Fall keinen langfristigen Zusammenhang zwischen Vergangenheit und Zukunft und das System lässt sich nur voraussagen, indem man es laufen lässt.

Das Computerprogramm «Simulator» unterstützt verschiedene Iterationsfunktionen, welche für gewisse Parameterwerte chaotisch werden. Dann lassen sich die Eigenschaften der Transitivität und Sensitivität konkret untersuchen und veranschaulichen.

Ebenso gibt es einen Modus «Zweidimensionale Darstellung». Dort werden Messungen eines Experimentators simuliert, welche im chaotischen Fall keine Rückschlüsse mehr auf das zugrundeliegende Bewegungsgesetz erlauben.

2.5. Histogramme

Um im chaotischen Fall einen Orbit zu untersuchen, kann man den Zustandsraum in disjunkte Intervalle aufteilen, deren Vereinigung wieder den Zustandsraum ergibt. Anschliessend zählt man bei

einem Intervall, wie oft es während der Iteration getroffen wird. So erhält man ein Histogramm und sieht, wie sich ein Orbit im Zustandsraum «verteilt».

Für einen Zustandsraum, der ein reelles Intervall $[a, b]$ ist, zerlegt der «Simulator» dieses in n gleichgroße Intervalle der Breite:

$$d := \frac{b - a}{n}$$

Dann sind diese Intervalle:

$$[a, a + d[\cup [a + d, a + 2d[\cup \dots \cup [a + (n - 1)d, b]$$

Anschliessend ermittelt der «Simulator», wie oft ein Intervall bei der Iteration von einem Punkt der Iteration getroffen wird, und zeigt das Histogramm an.

2.6. Protokolle

Angenommen, man hat zwei Intervalle $U, V \subset X$, $U \cap V = \emptyset$ und es gebe Bahnen (x_i) , welche ganz in $U \cup V$ verlaufen. Für eine solche Bahn erstellt man ein Protokoll wie folgt:

$$p(x_i) = \begin{cases} 0, & x_i \in U \\ 1, & x_i \in V \end{cases}$$

Beispiel

Beim 3n+1 Problem könnte man folgendes Protokoll p erstellen:

$$p(n) = \begin{cases} 1, & n \text{ ungerade} \\ 0, & n \text{ gerade} \end{cases}$$

Wie wir sehen werden, gibt es Beispiele von Iterationen, bei denen *jede* 0-1 Folge als Protokoll auftritt, wenn man den Startwert geeignet wählt. Damit hat man eine weitere Definition von «chaotisch»:

Ein dynamisches System heisst *chaotisch im Sinne des Münzwurfs*, wenn es zwei Mengen U und V wie eingangs beschrieben gibt, so dass jede 0-1 Folge als Protokoll einer Bahn vorkommt, wenn man den Startwert entsprechend wählt.

Sei $p = s_1 s_2 s_3 \dots s_n$, $s_i \in \{0,1\}$ ein Protokoll für die ersten n Schritte einer Iteration.

Dann ist in diesem Zusammenhang interessant, folgende Menge zu untersuchen:

$$A_p := \{x_o \in X \mid \text{Das Protokoll von } x_o \text{ beginnt mit } p\}$$

Es gilt:

$$A_{s_0} \supset A_{s_0 s_1} \supset A_{s_0 s_1 s_2} \supset \dots$$

In [2] wird gezeigt, dass obige Sequenz eine Intervallschachtelung ist, falls für die Iteration f und die Mengen U, V gilt:

- 1) $f(U) \supset U \cup V$ und ebenso $f(V) \supset U \cup V$
- 2) Es gibt eine Zahl $\lambda > 1$ so dass $|f'(x)| \geq \lambda, \forall x \in U \cup V$

Unter den Voraussetzungen 1) und 2) kann jedes Protokoll vorgegeben werden. Die obige Intervallschachtelung definiert dann den Startwert in $U \cup V$, der zum vorgegebenen Protokoll gehört.

Der «Simulator» ermöglicht im chaotischen Fall, ein beliebiges Protokoll vorzugeben (wegen der Rechengenauigkeit natürlich nur bis zu einer begrenzten Länge). Er schlägt dann einen Startwert für die Iteration vor, dessen Orbit genau das vorgegebene Protokoll liefert.

Einen Zusammenhang zwischen «*chaotisch im Sinne des Münzwurfs*» und der Definition im Sinne von Devaney wird in [2] hergestellt:

Angenommen, ein dynamisches System sei chaotisch im Sinne des Münzwurfs. Betrachte nun die Menge Σ aller 0-1 Folgen und die Menge der zugehörigen Startpunkte $\Lambda := \{x(s) | s \in \Sigma\}$. Dann ist das Subsystem (Λ, f) chaotisch (im Sinne von Devaney).

3. Beispiele diskreter dynamischer Systeme

Vorbemerkung: Auf die Zeltabbildung in Abschnitt 2 wird immer wieder Bezug genommen. Sie sollte nicht übergangen werden. Die übrigen Abschnitte in diesem Kapitel können unabhängig voneinander betrachtet werden.

3.1. Das Bernoulli-Shift-System

Als *Bernoulli-Shift-System* bezeichnet man folgendes diskretes dynamische System:

$$f: [0,1] \rightarrow [0,1]; f(x) = \begin{cases} 2x, & x \in [0,0.5[\\ 2x - 1, & x \in [0.5,1] \end{cases}$$

Dieses einfache System ist chaotisch.

Jede Hälfte des Intervalls $[0,1]$ wird zuerst mit dem Faktor 2 gestreckt. Anschliessend werden beide gestreckten Hälften übereinander geschoben. Das gibt eine «gute Durchmischung» der Punkte im Intervall. Die chaotischen Eigenschaften erkennt man, wenn man x in Dualbruchdarstellung betrachtet und dann die Wirkung der Abbildung untersucht. Sei $x_1 \in [0,1]$ ein Startpunkt der Abbildung, dargestellt als Dualbruch:

$$x_1 = 0.s_1s_2s_3\dots; s_i \in \{0,1\}$$

Dann wird dieser Dualbruch bei jedem Iterationsschritt durch die Multiplikation mit 2 um eine Stelle nach links geschoben, wobei die Stelle vor dem Komma verschwindet, und zwar wegen der Zuordnung $x \mapsto 2x - 1$ im Falle $x \geq 0.5$. Man erhält dann einen Orbit:

$$x_2 = 0.s_2s_3s_4\dots$$

$$x_3 = 0.s_3s_4\dots$$

Beispiel:

$$x_1 = 0.010010111$$

$$x_2 = 0.10010111$$

$$x_3 = 0.0010111$$

$$x_4 = 0.010111$$

$$x_5 = 0.10111$$

...

Die Sensitivität und Transitivität lassen sich leicht nachweisen:

Sensitivität

Wähle zu einem beliebigen Anfangswert

$$x_1 = 0.s_1s_2s_3\dots s_is_{i+1}s_{i+2}\dots$$

einen zweiten, bei dem die ersten i Stellen mit x_1 übereinstimmen und die weiteren Stellen komplementär sind (d.h. aus «0» wird eine «1» und umgekehrt):

$$x'_1 = 0.s_1s_2s_3\dots s_i\overline{s_{i+1}}\overline{s_{i+2}}\overline{s_{i+3}}\dots$$

Dann liegt x'_1 in einer beliebig kleinen Umgebung von x_1 :

$$x'_1 \in U_{2^{-i}}(x_1)$$

Aber nach i Schritten wird der Abstand der beiden Punkte maximal in $[0,1[$.

Transitivität

Sei ein Anfangspunkt

$$x_1 = 0, s_1 s_2 s_3 \dots s_i \dots$$

Und ein Zielpunkt

$$z = 0, t_1 t_2 t_3 \dots t_i \dots$$

Gegeben.

Wähle einen neuen Anfangspunkt:

$$x'_1 = 0, s_1 s_2 s_3 \dots s_i 000 \dots 000 t_1 t_2 t_3 \dots t_i \dots$$

x'_1 ist in einer beliebig kleinen Umgebung von x_1 in Abhängigkeit wie viele Nullen eingefügt werden.

Und nach i Schritten fällt x'_1 genau auf z .

Periodische Punkte

Jeder periodische Dualbruch liefert einen periodischen Orbit. Das sind genau die rationalen Zahlen im Intervall $[0,1[$ und diese liegen dicht in diesem Intervall. Jeder nicht-periodische Dualbruch führt zu einem aperiodischen Orbit.

Ferner geht bei jedem Schritt in Bit an Information über den Startwert verloren! Im Laufe der Iteration verschwindet diese Information vollständig.

Vorgegebene Protokolle

Wenn wir das Intervall aufteilen:

$$U = [0,0.5[, V = [0.5,1]$$

Dann gilt: $U \cap V = \emptyset$ und $U \cup V = [0,1]$

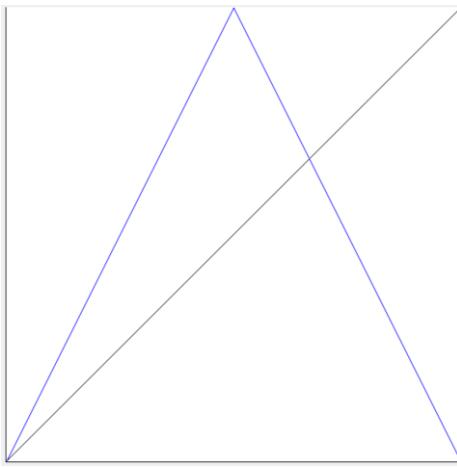
Wir erstellen ein Protokoll. Wir schreiben eine «0» wenn $x_i \in U$ und eine «1», wenn $x_i \in V$. Dann stimmt das Protokoll gerade mit der Dualdarstellung des Startwertes überein. Man hat also eine bijektive Zuordnung zwischen Startwerten und Protokollen. Insbesondere gibt es zu jedem beliebig vorgegebenen Protokoll einen Startwert, der dieses Protokoll liefert!

3.2. Die Zeltabbildung

Die *Zeltabbildung* ist eine Abwandlung des Bernoulli-Shift-Systems. Wir betrachten das Intervall $[0,1]$ und definieren die Zeltabbildung als:

$$z: [0,1] \rightarrow [0,1]; z(u) = \begin{cases} 2u, & u \in [0,0.5[\\ 2(1-u), & u \in [0.5,1] \end{cases}$$

Die Bezeichnungen wählen wir in Hinblick auf die Implementation im «Simulator».



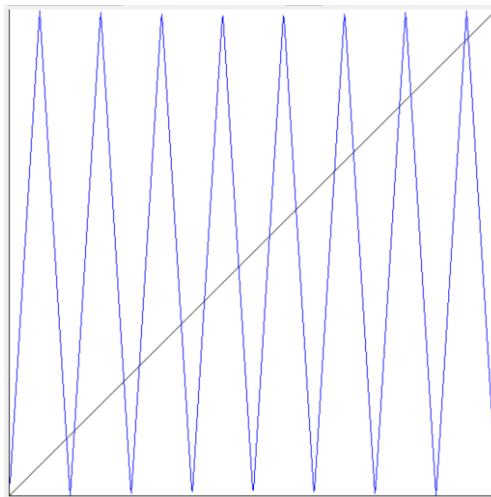
Graph der Zeltabbildung über dem Intervall $[0,1]$ erstellt durch den «Simulator»

Die Zeltabbildung schneidet die 45° Gerade in den Punkten 0 und $2/3$. Sie hat also die Fixpunkte

$$\xi_1 = 0 \text{ und } \xi_2 = 2/3$$

In diesen Punkten ist $|f'(\xi_1)| = |f'(\xi_2)| = 2 > 1$ und beide Punkte sind repulsiv.

Wenn man die Zeltabbildung iteriert, wird das Einheitsintervall bei jeder Iteration gestreckt und zusammengefaltet. Bei vier Iterationen zum Beispiel hat man den Graphen:



Graph der vierfach iterierten Zeltabbildung $f^4: [0,1] \rightarrow [0,1]$

Die oben dargestellte Funktion f^4 hat insgesamt 8 Fixpunkte (Schnittpunkte mit der 45° Geraden) und sämtliche sind repulsiv (die Ableitung an allen Fixpunktstellen ist > 1).

Wenn man weiter iteriert, verdoppeln sich bei jedem Iterationsschritt die Anzahl der Fixpunkte der iterierten Funktion. f^n hat 2^n repulsive Fixpunkte. Diese sind repulsive n-Zyklen der ursprünglichen Funktion f und liegen dicht im Intervall $[0,1]$.

Bei der Zeltabbildung lassen sich einige Sätze elementar beweisen. Wir nutzen die Gelegenheit. Diese Sätze sind aber für das Verständnis weiterer Kapitel nicht relevant.

Satz

Bei der Zeltabbildung ist jede rationale Zahl aus dem Einheitsintervall präperiodisch oder periodisch. Die zugehörigen Zyklen sind repulsiv. Sie liegen dicht im Einheitsintervall.

Beweis: $x = 1$ geht über in den Fixpunkt 0. Eine rationale Zahl aus dem Einheitsintervall, die kleiner 1 ist, hat die Form $x = \frac{p}{q}$ mit $1 \leq p < q$ und wir können annehmen, dass der Bruch gekürzt ist.

Wenn $x < 0.5$, also $2p < q$, dann geht x über in $\frac{2p}{q}$. Wenn $x \geq 0.5$, also $2p \geq q$, dann geht x über in $\frac{2q-2p}{q}$. Wenn der Nenner q den Faktor 2 enthält, wird dieser gekürzt und im Laufe der Iteration wird der Faktor 2 im Nenner verschwinden. $x = \frac{1}{2^k}$ geht im Laufe der Iteration nach 0. Wenn q andere Faktoren als 2 enthält, dann wird x im Laufe der Iteration immer durch einen gekürzten Bruch der Form $\frac{1}{q'}$ dargestellt. Da es nur endlich viele Zähler der gekürzten Brüche mit Nenner q' gibt, muss im Laufe der Iteration ein Bruch auftreten, der bereits früher aufgetreten ist. Dort beginnt der Zyklus.

Die rationalen Zahlen liegen ferner dicht im Einheitsintervall. Da $|f'(x)| = 2$ in $[0,1] \setminus \{0.5\}$ sind alle Zyklen repulsiv. Der Punkt 0.5 geht über in den Repellor 0. \square

Satz

Die Zeltabbildung besitzt für jedes $n \in \mathbb{N}$ einen echten Zyklus der Länge n .

Beweis: Betrachte den Startwert $x_1 = \frac{2}{2^n+1}$

Dieser wird bei jedem Schritt mit 2 multipliziert, solange das Resultat < 0.5 ist. Das ist der Fall, bis zum Wert

$$x_{n-1} = \frac{2^{n-1}}{2^n + 1}$$

Wie man leicht prüft, ist noch immer $x_{n-1} < \frac{1}{2}$. Beim nächsten Schritt erhält man:

$$x_n = \frac{2^n}{2^n + 1} > \frac{1}{2}$$

Also wird gemäß Iterationsvorschrift

$$x_{n+1} = 2(1 - x_n) = 2 \frac{1}{2^n + 1} = x_1$$

Alle auftretenden Brüche lassen sich nicht kürzen, also hat der Zyklus effektiv die Länge n . \square

Satz

Die Zeltabbildung ist sensitiv.

Beweis: Wenn wir eine Zahl $x \in [0,1]$ als Dualbruch darstellen, dann sehen wir die Wirkung der Zeltabbildung ähnlich wie beim Bernoulli Shift.

Es gilt in dieser Darstellung und wenn $s_i \in \{0,1\}$ die i-te duale Ziffer nach dem Komma ist:

$$f: \begin{cases} 0,0s_2s_3s_4 \dots \mapsto 0, s_2s_3s_4 \dots \\ 0,1s_2s_3s_4 \dots \mapsto 0, \bar{s}_2\bar{s}_3\bar{s}_4 \dots \end{cases}$$

Dabei bedeuten \bar{s}_i das duale Komplement von s_i .

Sei nun ein Startwert in Dualdarstellung gegeben:

$$x_1 = 0.s_1 s_2 s_3 s_4 \dots$$

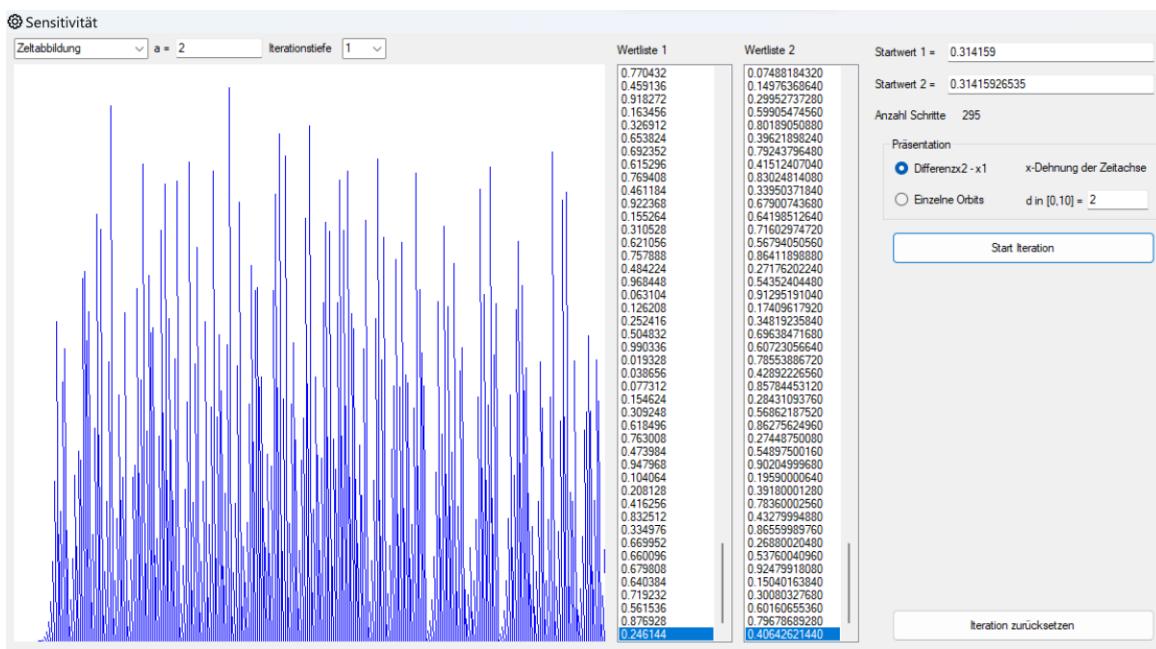
Wir schneiden dann diesen Dualbruch nach einer genügend grossen Anzahl Stellen ab (oder ergänzen fehlende Stellen mit «0»), damit der so geänderte Startwert nahe genug beim Ursprünglichen ist.

Wenn darin eine ungerade Zahl «1» vorkommt, wählen wir $x'_1 = 0.s_1 s_2 s_3 s_4 .. 00 \dots s_1 s_2 s_3 s_4 \dots$. Das heisst, wir hängen dann nochmals die Ziffern von x_1 an den Schluss an. Bei jedem Iterationsschritt rücken diese Stellen um eine Position nach links, wobei bei jeder auftretenden «1» an erster Position hinter dem Komma das duale Komplement der restlichen Stellen gebildet wird. Da dies eine ungerade Anzahl mal vorkommt, steht nach entsprechend vielen Iterationsschritten die Zahl da: $x_n = 0.\bar{s}_1 \bar{s}_2 \bar{s}_3 \dots$, also eine Zahl mit maximalem Abstand zu x_1 .

Analog gehen wir vor, wenn der abgeschnittene und allenfalls mit «0» ergänzte Dualbruch von x_1 eine gerade Anzahl «1» enthält, und setzen $x'_1 = 0.s_1 s_2 s_3 s_4 .. 00 \dots \bar{s}_1 \bar{s}_2 \bar{s}_3 \bar{s}_4 \dots$ mit demselben Resultat. □

Der «Simulator» nutzt diese Methode, um die Sensitivität der Zeltabbildung darzustellen im Menüpunkt «Unimodale Funktionen – Sensitivität». Man kann zwei sehr nahe beieinanderliegende Startwerte wählen, und dann wahlweise

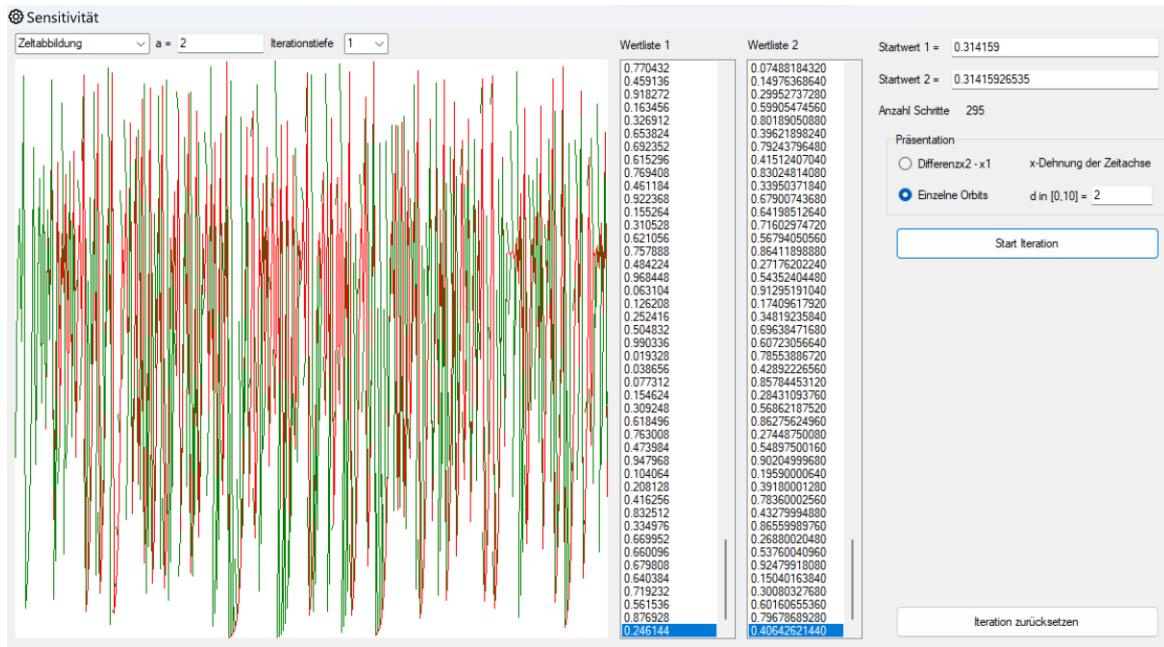
- a) Die Differenz der beiden erzeugten Orbits im Laufe der Iteration (auf der «Zeitachse» beim «Simulator») darstellen:



Differenz zweier Orbits mit nahe beieinanderliegenden Startwerten der Zeltabbildung

oder

- b) jeden einzelnen Orbit auf der Zeitachse darstellen und die Orbits vergleichen



Unterschiedliche Orbits zweier nahe beieinanderliegender Startwerte bei der Zeltabbildung

Die beiden Startwerte unterscheiden sich nur in ab der 7-ten Stelle hinter dem Komma, aber nach wenigen Schritten beginnen die Orbits auseinanderzulaufen. Ihre Differenz erreicht zeitweise fast die gesamte Intervallbreite.

Satz

Die Zeltabbildung ist transitiv.

Beweis:

Wie beim Nachweis der Sensitivität gehen wir von einem Startwert $x_1 = 0.s_1s_2s_3s_4 \dots$ und einem Zielwert $z_1 = 0.t_1t_2t_3 \dots$ aus, beide in Dualbruchdarstellung. Wir schneiden den Dualbruch von x_1 wieder nach genügend vielen Stellen ab (oder ergänzen fehlende Stellen mit «0»), damit der geänderte Startwert nahe genug beim Ursprünglichen ist. Wenn der verbleibende Dualbruch eine gerade Anzahl «1» enthält, hängen wir die Stellen des Zielwertes hinten an diesen an. Wenn die Anzahl «1» ungerade ist, dann hängen wir das duale Komplement hinten an. In beiden Fällen erhalten wir nach genügend vielen Iterationsschritten $z_n = 0.t_1t_2t_3 \dots \square$

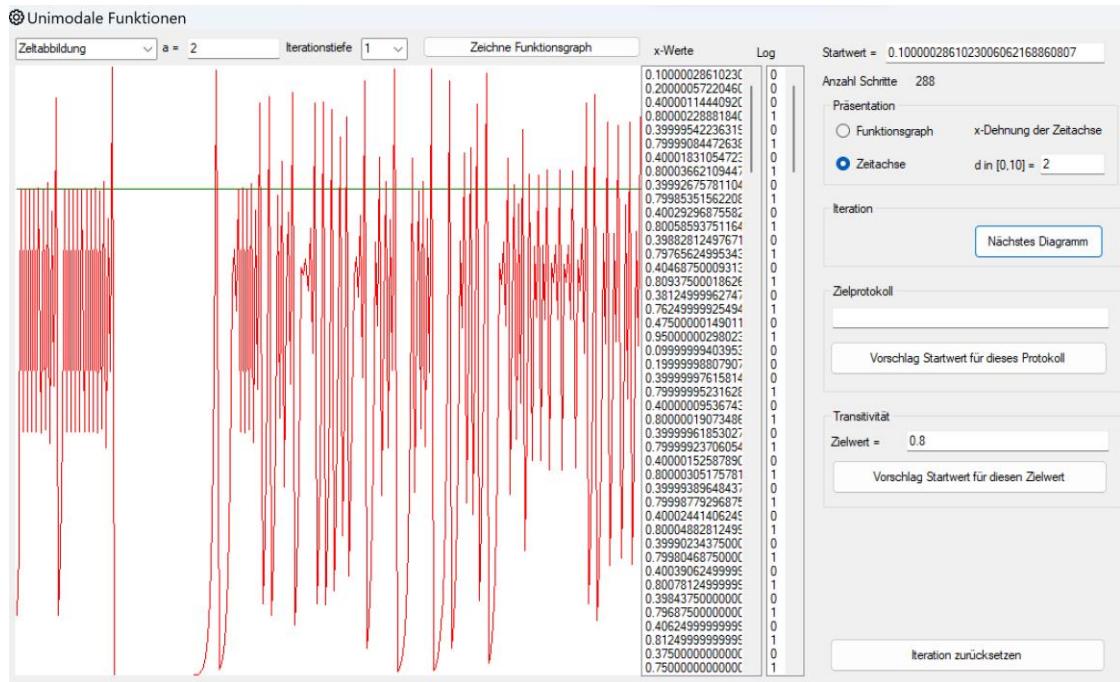
Es ist leicht einzusehen, dass die Zyklen durch periodische Dualbrüche dargestellt werden und dass diese dicht in $[0,1]$ liegen. Somit gilt:

Satz

Die Zeltabbildung ist chaotisch.

Für die Darstellung der Transitivität benutzt der «Simulator» dieselbe Methode wie beim vorherigen Beweis, um einen leicht abgeänderten Startwerte vorzuschlagen, welcher dem vorgegebenen Zielwert nahekommt. Der abgeänderte Dualbruch wird dann in einen Dezimalbruch umgerechnet. Wegen der endlichen Stellenzahl und weil mit der Zeit sämtliche Information über den Anfangswert verschwindet, wird die Iteration wieder irgendwelche Werte produzieren, nachdem man dem Zielwert nahegekommen ist.

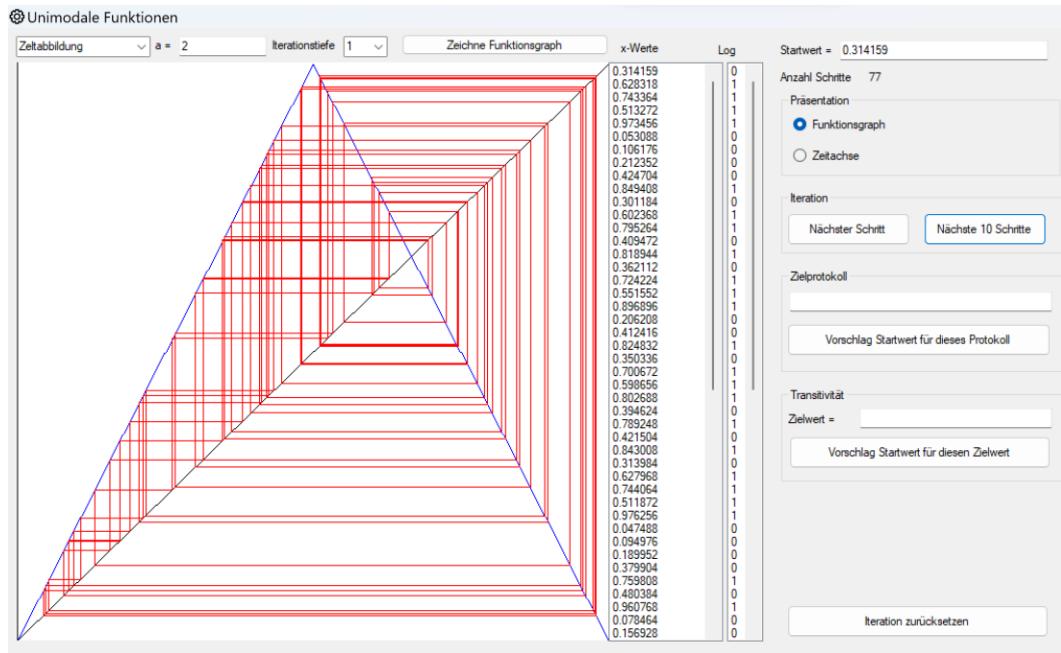
Man findet diese Funktion im Menüpunkt «Unimodale Funktionen – Iteration».



Die Iteration wird auf der Zeitachse dargestellt. Grüne Linie: Der Zielwert

Im obigen Bild wurde der Startwert 0.1 und der Zielwert 0.8 vorgegeben. Der «Simulator» hat dann für diesen Zielwert einen neuen Startwert berechnet, der nahe beim ursprünglichen ist, nämlich 0.1000002861023006062168860807. Schon nach 4 Iterationsschritten kommt man dem Zielwert sehr nahe.

Protokolle



Iteration der Zeltabbildung dargestellt im Funktionsgraph

Der «Simulator» iteriert in obiger Abbildung die Zeltabbildung mit den ersten Ziffern von π als Startwert. Rechts von Graphen sind die iterierten Werte aufgelistet.

Für die Zeltabbildung definieren wir ein Protokoll p nach folgender Vorschrift:

$$p(x_n) = \begin{cases} 0, & x_n \in [0, 0.5] \\ 1, & x_n \in [0.5, 1] \end{cases}$$

Dieses Protokoll ist im obigen Bild in der Spalte «log» aufgelistet.

Wie beim Bernoulli Shift kann für jedes vorgegebene Protokoll ein Startwert gefunden werden, der dieses Protokoll liefert. Um von diesem vorgegebenen Protokoll zum Startwert zu gelangen, rechnet der «Simulator» die Protokollierung rückgängig:

Man beginnt mit der letzten Stelle im Protokoll. Dann ist der weitere Algorithmus der folgende:

Wenn im Protokoll eine «0» auftaucht, dann schreibe eine «0» hinter das Komma und schiebe alle übrigen Stellen um eine Position nach rechts.

Wenn im Protokoll eine «1» auftaucht, dann schreibe eine «1» hinter das Komma, bilde von allen übrigen Stellen das Komplement und schiebe sie um eine Position nach rechts.

Beispiel

Betrachte das Protokoll $p = 10110$. Wenn wir das in 5 Schritten von rechts nach links abarbeiten, erhalten wir einen Startwert mit 5 Ziffern nach dem Komma.

Wir beginnen mit der hintersten Position «0». Die fünfte Ziffer des Startwertes ist also eine «0».

$$x_1 = 0, s_1 s_2 s_3 s_4 0$$

Die nächste Position im Protokoll ist die «1». Wir notieren also eine «1» an der vierten Stelle im Startwert und bilden von den Ziffern rechts davon das duale Komplement.

$$x_1 = 0, s_1 s_2 s_3 11$$

Wir fahren fort mit der nächsten Position im Protokoll, einer «1»:

$$x_1 = 0, s_1 s_2 100$$

Die «0» an Position 2 im Protokoll liefert:

$$x_1 = 0, s_1 0100$$

Und schliesslich die «1» an erster Stelle:

$$x_1 = 0,11011$$

Man erhält also einen Startwert $x_1(s_1 s_2 s_3 s_4 \dots)$ für ein Protokoll $p=0.s_1 s_2 s_3 s_4 \dots$ in Dualdarstellung. Wenn das Protokoll die Länge n hat, dann liegen alle Startwerte, welche das vorgegebene Protokoll liefern, im Intervall $[x_1(p), x_1(p) + 2^{-(n+1)}[$. Damit Rundungseffekte des Computers weniger ins Gewicht fallen, wählen wir den Startwert aus der Intervallmitte: $\tilde{x}_1(p) := x_1(p) + 2^{-(n+2)}$.

Zur Funktion für das vorgegebene Protokoll gelangt man im Menü «Unimodale Funktionen – Iteration».

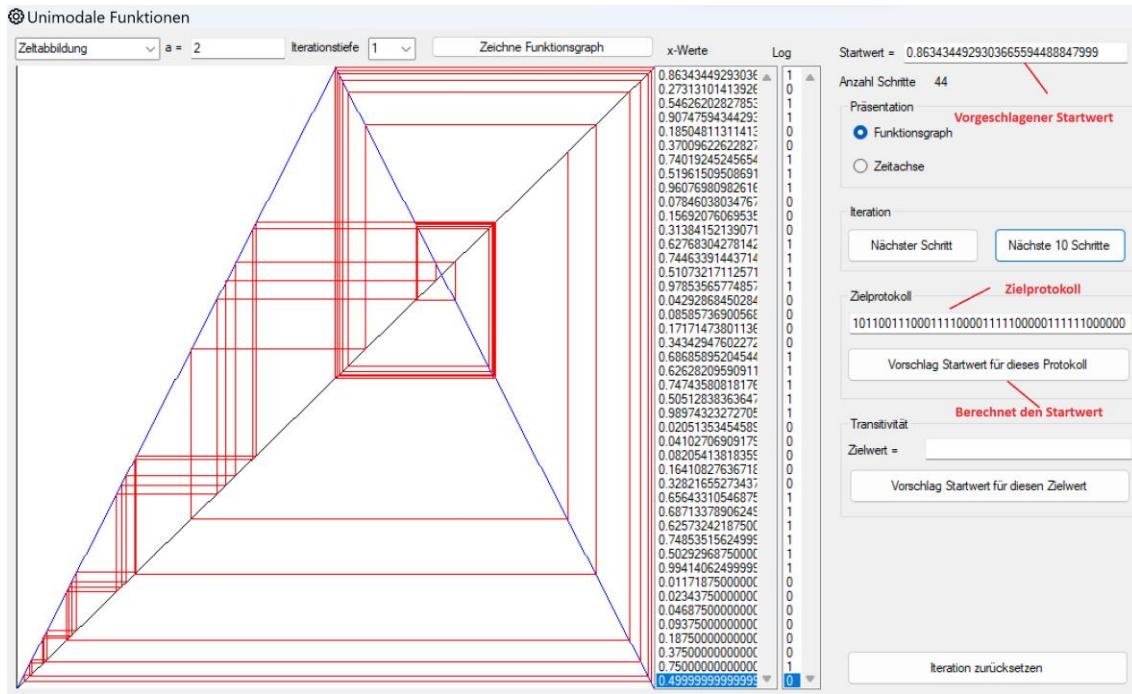
Wir geben im folgenden Beispiel als Zielprotokoll ein:

101100111000111100001111100000111111000000

Dieses Zielprotokoll hat die Länge 42. Mit der Taste «Startwert vorschlagen» wird dann der Startwert als Dezimalbruch vorgeschlagen, nämlich:

0.8634344929303665594488847999

Wie man sieht, entsteht bei der Iteration das gewünschte Protokoll:



Zeltabbildung mit vorgegebenem Protokoll der Länge 42

Zeltabbildung mit einem Parameter a

Als Variante der Zeltabbildung kann man die Abbildung betrachten:

$$z_a: [0,1] \rightarrow [0,1]; z_a(u) = \begin{cases} au, u \in [0,0.5[\\ a(1-u), u \in [0.5,1] \end{cases}, a \in]0,2]$$

Diese hat den Fixpunkt $u = 0$ und dieser ist attraktiv für $a < 1$.

Für $u \geq 0.5$ kommt als Fixpunkt $u = \frac{a}{1+a}$ in Frage. Damit $\frac{a}{1+a} \geq 0.5$ ist, muss $a \geq 1$ sein. Der Fixpunkt $u = \frac{a}{1+a}$ ist also repulsiv.

Für die Ableitung der n -fach iterierten Funktion gilt: $|z_a^{n'}(u)| = a$. Somit sind für $a > 1$ alle potenziellen n -Zyklen repulsiv.

Es gibt für $a > 1$ keine attraktiven Zyklen und für $a < 1$ nur den (attraktiven) Fixpunkt 0.

Deshalb betrachtet man in der Regel nur die eingangs definierte Zeltabbildung mit $a = 2$. Der «Simulator» unterstützt aber das Experimentieren für $a \in]0,2]$.

3.3. Das logistische Wachstum

Wir betrachten ein einfaches System, um das Wachstum einer Bevölkerung zu simulieren. Das Ziel ist nicht, ein realistisches Modell für existierende Populationen zu entwerfen, sondern zu zeigen, dass bereits ein sehr vereinfachtes Modell unter gewissen Bedingungen chaotische Eigenschaften haben kann. Das heisst, das «Chaos» ist nicht in der Komplexität eines Modells begründet, sondern tritt schon bei sehr einfachen Modellen auf, wie wir auch bei der Zeltabbildung gesehen haben.

Wir nehmen an, die Grösse dieser Population liege im Intervall $[0, 1]$. 0 heisst, dass sie ausgestorben ist und 1, dass sie die theoretisch möglichen 100% erreicht hat. Nun bezeichnen wir die Grösse der Population der n-ten Generation mit $x_n \in [0,1], n \in \mathbb{N}$.

Die nächste Generation x_{n+1} wird einerseits grösser, je grösser x_n war, auf Grund der höheren Reproduktionszahl. Wenn die Population aber an ihre ressourcenbedingt natürlichen Grenzen stösst, dann kann es sein, dass die Population wieder kleiner wird, weil nicht mehr alle Individuen überleben können. Dieser Zusammenhang beschreiben wir durch folgende Iterationsvorschrift:

$$x_{n+1} = f(x_n) := ax_n(1 - x_n) \text{ wobei } x_n \in [0,1], \forall n \in \mathbb{N} \text{ und } a \in]0,4]$$

Der Parameter a kann aufgefasst werden als Steuerung der Reproduktionsrate. Diese Iterationsvorschrift ist bekannt als *logistisches Wachstum*.

Wir wollen zuerst untersuchen, welche Werte von a sinnvoll sind. Der Fall $a = 0$ ist trivial.

Für $x_n \in [0,1]$ ist $x(1 - x)$ positiv, also muss a ebenfalls positiv sein, damit der Funktionswert positiv ist. Andererseits nimmt die Funktion f bei $x = 0.5$ ein Maximum an und es ist:

$$f(0.5) = \frac{a}{4}$$

Damit der Funktionswert wieder im Intervall $[0,1]$ liegt, muss gelten $a \leq 4$. Daraus ergibt sich schliesslich die Bedingung $a \in]0,4]$. In Zukunft schreiben wir für das logistische Wachstum:

$$f: [0,1] \rightarrow [0,1], x \mapsto ax(1 - x), a \in]0,4]$$

Wir untersuchen zuerst, welche Attraktoren es gibt.

Periodische Punkte

ξ ist ein Fixpunkt der logistischen Wachstumsfunktion, falls gilt: $f(\xi) = \xi$ bzw. $a\xi(1 - \xi) = \xi$

Somit hat man zwei Lösungen: $\xi_1 = 0$ und $\xi_2 = 1 - 1/a$

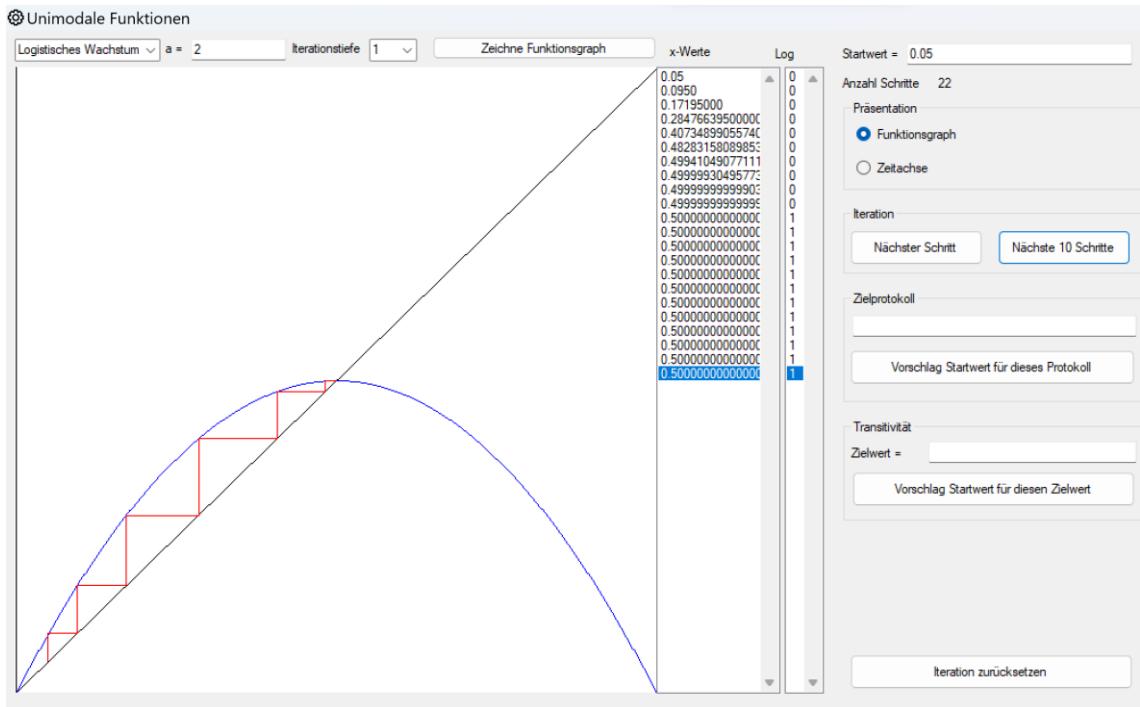
Ob diese Fixpunkte attraktiv oder repulsiv sind, hängt von ihrem Multiplikator, bzw. vom Wert der Ableitung im Betrag an deren Stelle ab. Es ist:

$$\lambda = |f'(x)| = |a(1 - 2x)|$$

Der erste Multiplikator ist $\lambda_1 = |f'(0)| = a$. Also ist der erste Fixpunkt $\xi_1 = 0$ attraktiv für $a < 1$ und repulsiv für $a > 1$. Ferner ist:

$$\lambda_2 = |f'(1 - 1/a)| = |2 - a| < 1 \text{ für } a \in]1,3[$$

Der Fixpunkt $\xi_2 = 1 - 1/a$ ist attraktiv für $a \in]1,3[$ und repulsiv für $a \in]3,4[$.



Logistisches Wachstum für $a = 2$ mit attraktivem Fixpunkt $\xi_2 = 0.5$

Zyklen der Periode 2 sind Fixpunkte von f^2 . $f^2(\xi) = \xi$ liefert eine Gleichung vierten Grades, wobei wir bereits zwei Nullstellen kennen, nämlich die Fixpunkte von f : $\xi_1 = 0$ und $\xi_2 = 1 - 1/a$.

Um die anderen beiden Fixpunkte ξ_3 und ξ_4 zu finden verwenden wir den Ansatz:

$$\begin{cases} \xi_3 = f(\xi_4) = a\xi_4(1 - \xi_4) \\ \xi_4 = f(\xi_3) = a\xi_3(1 - \xi_3) \end{cases}$$

Wenn man die untere Gleichung von der oberen subtrahiert, erhält man:

$$\xi_3 - \xi_4 = -a(\xi_3 - \xi_4) + a(\xi_3^2 - \xi_4^2)$$

Da die gesuchten Fixpunkte echt 2-periodisch sind, gilt $\xi_3 \neq \xi_4$ und wir erhalten:

$$1 = -a + a(\xi_3 + \xi_4)$$

Wir setzen $\xi_4 = \frac{1+a}{a} - \xi_3$ in die zweite Gleichung ein und erhalten nach etwas Rechnung:

$$a^2\xi_3^2 - a(1 + a)\xi_3 + 1 + a = 0$$

Das liefert:

$$\xi_{3,4} = \frac{1 + a \pm \sqrt{(1 + a)(a - 3)}}{2a}$$

Für $a > 3$ erhalten wir also einen 2-Zyklus.

Dieser ist attraktiv, wenn

$$|f''(\xi_3)| = |f'(f(\xi_3) \cdot f'(\xi_3))| = |f'(\xi_4) \cdot f'(\xi_3)| < 1$$

Es ist:

$$f'(x) = a(1 - 2x)$$

$$|f'(\xi_4) \cdot f'(\xi_3)| = a^2 |1 - 2(\xi_3 + \xi_4) + 4\xi_3\xi_4|$$

Mit

$$\xi_3 + \xi_4 = \frac{1+a}{a} \text{ und } \xi_3 \xi_4 = \frac{1+a}{a^2}$$

Erhalten wir für den Multiplikator des 2-Zyklus:

$$\lambda_{3,4} = |f'(\xi_4) \cdot f'(\xi_3)| = |a^2 - 2a - 4|$$

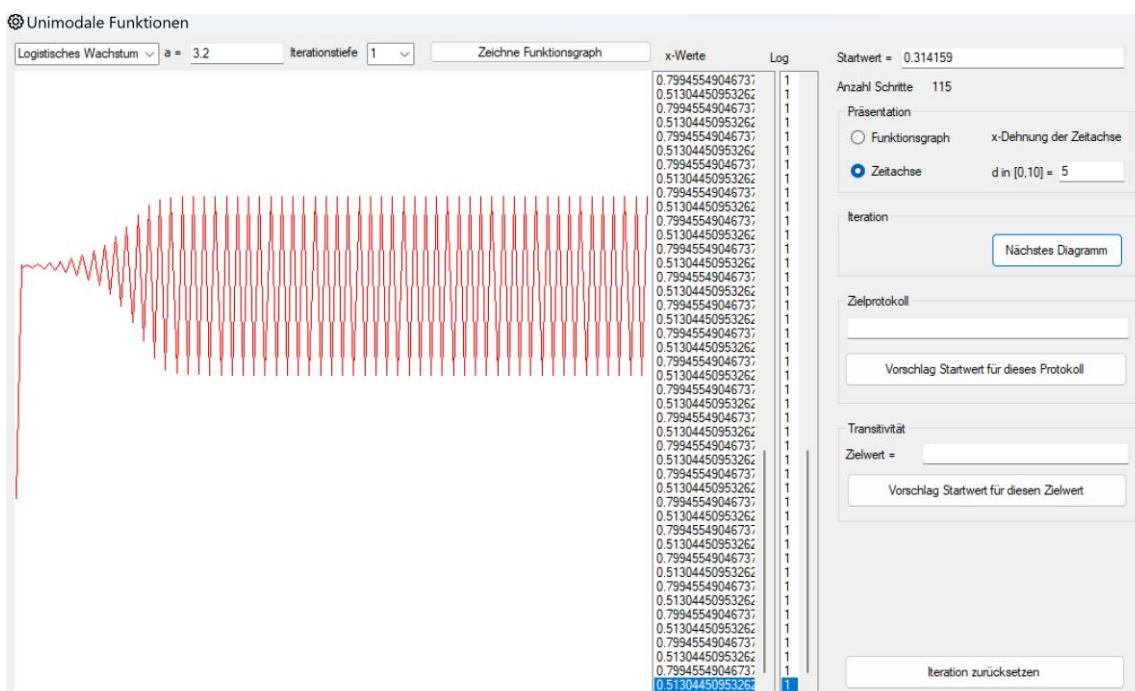
Der 2-Zyklus ist attraktiv für $\lambda_{3,4} < 1$ und wir untersuchen die Übergangsstellen:

$$a^2 - 2a - 4 = \pm 1$$

Diese Gleichung hat die Lösungen $a = 1 \pm \sqrt{6}$ wobei wegen $a > 0$ nur in Frage kommt:

$$a = 1 + \sqrt{6} \approx 3.449499$$

Der 2-Zyklus ist also attraktiv im Intervall $a \in]3, 3.449499[$



Logistisches Wachstum für $a = 3.2$, dargestellt auf der Zeitachse

Im obigen Bild ist $a = 3.2$. Die Folge pendelt sich bald in den attraktiven 2-Zyklus ein:

$$\xi_{3,4} = \begin{cases} 0.799455 \dots \\ 0.513044 \dots \end{cases}$$

Allfällige weitere Zyklen sind Fixpunkte von f^k , also Lösungen von Gleichungen vom Grad 2^k . Die Fixpunkte von f^k sind die Schnittpunkte des Graphen von f^k mit der 45° -Geraden.

3.4. Konjugierte der Zeltabbildung

Die Zeltabbildung ist definiert als:

$$z: [0,1] \rightarrow [0,1]; z(u) = \begin{cases} 2u, & u \in [0,0.5[\\ 2(1-u), & u \in [0.5,1] \end{cases}$$

Nun sei $[a, b]$ ein reelles Intervall und $T: [0,1] \rightarrow [a, b]$ ein Diffeomorphismus. Das heisst, dass sowohl T wie auch die Umkehrabbildung T^{-1} stetig differenzierbar sind.

Dann nennen wir die Abbildung f :

$$f := T \circ z \circ T^{-1}: [a, b] \rightarrow [a, b]$$

Eine *Konjugierte* der Zeltabbildung z .

Beispiel

Das logistische Wachstum ist für $a = 4$ eine Konjugierte der Zeltabbildung. Die zugehörige Transformation ist gegeben durch:

$$T: [0,1] \rightarrow [0,1], u \in [0,1] \mapsto x = T(u) = \sin^2 \frac{\pi}{2} u \in [0,1]$$

T ist bijektiv, stetig differenzierbar und ebenso ist T^{-1} stetig differenzierbar.

Sei f das logistische Wachstum. Wir zeigen: $f \equiv T \circ z \circ T^{-1}$ auf $[0, 1]$. Sei $u \in [0, 1]$. Dann gilt:

$$\begin{aligned} f(T(u)) &= f\left(\sin^2 \frac{\pi}{2} u\right) = 4 \sin^2 \frac{\pi}{2} u \left(1 - \sin^2 \frac{\pi}{2} u\right) = (2 \sin \frac{\pi}{2} u \cdot \cos \frac{\pi}{2} u)^2 = \sin^2 \frac{\pi}{2} \cdot 2u \\ &= \begin{cases} \sin^2 \frac{\pi}{2} \cdot 2u, & u \in [0,0.5[\\ \sin^2 \frac{\pi}{2} \cdot 2(1-u), & u \in [0.5,1] \end{cases} = \begin{cases} \sin^2 \frac{\pi}{2} \cdot 2u \\ \sin^2(\pi - \frac{\pi}{2} \cdot 2u) \end{cases} = T(z(u)) \end{aligned}$$

Beachte: $\sin(\pi - \alpha) = \sin \alpha$

Somit ist auf $[0, 1]$: $f \circ T \equiv T \circ z \square$

Eine Konjugierte der Zeltabbildung übernimmt deren Eigenschaften hinsichtlich periodischem und chaotischem Verhalten.

Satz

Sei f eine Konjugierte der Zeltabbildung z unter dem Diffeomorphismus T . Dann gilt:

- 1) $f^n = T \circ z^n \circ T^{-1}, \forall n \in \mathbb{N}$ d.h. f^n ist eine Konjugierte von z^n
- 2) $\{u_1, u_2, \dots, u_k\}$ ist ein k -Zyklus von $z \Leftrightarrow \{T(u_1), T(u_2), \dots, T(u_k)\}$ ist ein k -Zyklus von f
- 3) Der k -Zyklus von z und der entsprechende k -Zyklus von f haben denselben Multiplikator
- 4) Jeder k -Zyklus von f ist repulsiv. Die Menge dieser Zyklen liegt dicht in $[0,1]$
- 5) f ist sensitiv und transitiv
- 6) f hat chaotische Eigenschaften

Somit ist das logistische Wachstum für $a = 4$ chaotisch.

Beweis des Satzes

$$1) f^n = (T \circ z \circ T^{-1})^n = T \circ z \circ T^{-1} \circ T \circ z \circ T^{-1} \circ \dots \circ T \circ z \circ T^{-1} = T \circ z^n \circ T^{-1}$$

2) $u \in [0, 1]$ ist ein Fixpunkt von $z^k \Leftrightarrow T(u)$ ist ein Fixpunkt von f^k :

$$\Rightarrow": z^k(u) = u \Rightarrow f^k(T(u)) = (T \circ z^{k \circ T^{-1}})(T(u)) = T(z^k(u)) = T(u)$$

$$\Leftrightarrow T(z^k(u)) = T(u) \Rightarrow z^k(u) = u$$

3) Sei $\xi = T(u)$ für ein $u \in]0,1[$ und u ein Fixpunkt von z^k , also ein Zyklus von z (der Fall $u = 0$ und $u = 1$ ist trivial). Es gilt also $z^k(u) = u$. Dann ist ξ ein Fixpunkt von $f^k: f^k(\xi) = \xi$.

Beachte ferner: $T(T^{-1}(\xi)) = \xi$, also $[T(T^{-1}(\xi))]' = T'(u) \cdot T^{-1}'(\xi) = 1, u \in]0,1[$ nach dem Satz über die Ableitung der Umkehrabbildung.

Dann gilt für die Multiplikatoren nach der Kettenregel:

$$\begin{aligned}\lambda_\xi &= |f^k'(\xi)| = |(T \circ z^{k \circ} T^{-1})'(\xi)| = |T'(z^k(T^{-1}(\xi))) \cdot z^{k'}(T^{-1}(\xi)) \cdot T^{-1}'(\xi)| = \\ &= |T'(z^k(u)) \cdot z^{k'}(u) \cdot T^{-1}'(\xi)| = |T'(u) \cdot z^{k'}(u) \cdot T^{-1}'(\xi)| = |z^{k'}(u)| = \lambda_u\end{aligned}$$

4) Folgt direkt aus 3) da alle Zyklen der Zeltabbildung repulsiv sind.

Wegen der Stetigkeit von T liegen die Zyklen von f dicht in $[0, 1]$. Wir wollen auf die ε -Begründung verzichten und den Beweis lediglich skizzieren: Sei $\xi \in [0, 1]$ und $u = T^{-1}(\xi)$. Dann gibt es in beliebiger Nähe von u einen zyklischen Punkt v . Da T stetig ist, liegt der ebenfalls zyklische Punkt $T(v)$ beliebig nahe bei $\xi = T(u)$.

5) Beweisskizze:

Transitivität: Sei $\xi_1 \in [0, 1]$ ein Startwert und $u_1 = T^{-1}(\xi_1)$. Sei $\xi \in [0, 1]$ ein Zielwert und $v = T^{-1}(\xi)$. Wegen der Transitivität von z gibt es in beliebiger Nähe von u_1 einen Punkt u_2 der im Laufe der Iteration v beliebig nahe kommt. Wegen der Stetigkeit von T kommt dann $T(u_2)$ dem Zielpunkt $\xi = T(v)$ beliebig nahe.

Sensitivität: T ist ein Diffeomorphismus. Insbesondere ist die Umkehrfunktion T^{-1} ebenfalls stetig.

Sei nun $\xi_1, \xi_2 \in [0, 1]$ beliebig nahe beieinander. Dann sind bei geeigneter Nähe dieser Punkte wegen der Stetigkeit von T^{-1} auch die Punkte $u_1 = T^{-1}(\xi_1), u_2 = T^{-1}(\xi_2)$ beliebig nahe beieinander. Im Laufe der Iteration der Zeltabbildung kann der Abstand dieser Punkte innerhalb des Intervalls $[0, 1]$ beliebig wachsen. Insbesondere gibt es ein $n \in \mathbb{N}$ so dass $z^n(u_1)$ nahe bei 0 und $z^n(u_2)$ nahe bei 1 ist (oder umgekehrt, o.B.d.A.). Wegen der Stetigkeit von $T: [0, 1] \rightarrow [a, b]$ ist dann $f^n(\xi_1) = T(z^n(u_1))$ nahe bei a und $f^n(\xi_2) = T(z^n(u_2))$ nahe bei b (oder umgekehrt).

Bemerkung

Dieser Ansatz ist ein einfaches Beispiel eines allgemeineren Sachverhaltes. Wir haben eine Transformation $T(u) = \sin^2 \frac{\pi}{2} u, u \in [0, 1]$ gefunden, so dass im Wesentlichen gilt:

$$f(T(u)) = T(2u)$$

Wenn wir in den Bereich der komplexen Zahlen gehen, dann gilt ein Satz, der auf Henri Poincaré zurückgeht und als Poincaré Funktionalgleichung bekannt ist:

Sei $f(z), z \in \mathbb{C}$ eine in $z = 0$ holomorphe Funktion mit $f(0) = 0, \lambda := f'(0), |\lambda| > 1$. Dann gibt es eine in $z = 0$ holomorphe Funktion $T(z), T(0) = 0$, so dass gilt:

$$f(T(z)) = T(\lambda z)$$

Wenn zusätzlich verlangt wird, dass $T'(0) = 1$, dann ist T eindeutig bestimmt. T heißt Poincaré Funktion.

Wenn eine Iteration der Form $x_{n+1} = f(x_n)$ definiert ist, gilt unter obigen Voraussetzungen:

$$x_n = T(\lambda^n c)$$

Dabei ist $c \in \mathbb{C}$ eine Konstante, welche vom Startwert x_0 abhängt.

Die Poincaré Funktion ist allerdings nur in wenigen speziellen Fällen, wie etwa für die logistische Funktion mit $a = 4$, eine elementare Funktion.

Protokolle

Im «Simulator» erstellen wir ein Protokoll der durch das logistische Wachstum erzeugten Zahlenfolge. Und zwar setzen wir als Protokoll p

$$p(x_n) = \begin{cases} 0, & \text{falls } x_n \in [0, 0.5[\\ 1, & \text{falls } x_n \in [0.5, 1] \end{cases}$$

Im chaotischen Fall $a = 4$ ist das logistische Wachstum eine Konjugierte der Zeltabbildung. Für die Transformation gilt dann: $T(0.5) = \sin^2 \frac{\pi}{4} = 0.5$ und T monoton wachsend. Somit gilt:

Die Protokolle des logistischen Wachstums für $a = 4$ und der Zeltabbildung stimmen bei der Iteration überein!

Bei der Zeltabbildung haben wir eine Bijektion zwischen Protokollen und Startwerten gefunden durch die Angabe einer Umkehrabbildung, welche jedem beliebigen Protokoll einen Startwert zuordnet, welcher bei der Iteration dieses Protokoll ergibt.

Damit erhalten wir auch eine Umkehrfunktion für die Protokollierung des logistischen Wachstums:

Ausgehend von einem beliebigen Protokoll berechnen wir den entsprechenden Startwert bei der Zeltabbildung, welcher bei der Iteration dieses Protokoll ergibt. Diese Berechnung haben wir im Abschnitt über die Zeltabbildung bereits explizit beschrieben. Dann verwenden wir wieder die Transformation

$$x = \sin^2 \frac{\pi}{2} u$$

Um daraus den entsprechenden Startwert für das logistische Wachstum zu berechnen. Dieser liefert bei der Iteration genau das vorgegebene Protokoll.

Beispiel

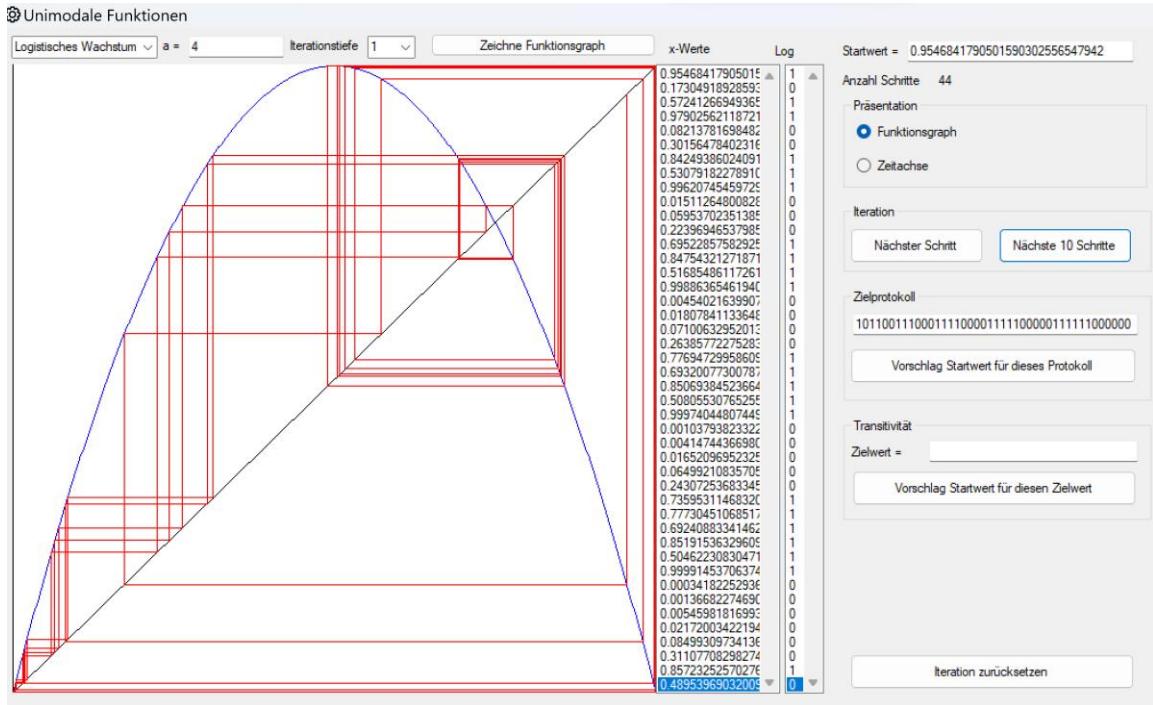
Wir geben als Zielprotokoll wieder vor:

1011000111000111100001111100000111111000000

Der durch den «Simulator» vorgeschlagene Startwert

0.9546841790501590302556547942

Liefert dann das Zielprotokoll. Wichtig dabei ist, dass der Parameter $a = 4$ ist, sonst liegt u.U. kein chaotisches Verhalten vor und der im «Simulator» implementierte Algorithmus ist auch nur für den Fall $a = 4$ gedacht.



Logistisches Wachstum mit vorgegebenem Protokoll der Länge 42

3.5. Die «normierte» Parabel

Oft wird in der Literatur an Stelle der logistischen Wachstumsfunktion die Parabel

$$g(y) = 1 - \mu y^2, [-1,1] \rightarrow [-1,1], \mu \in]0,2]$$

und ihre Iteration diskutiert. Zur Unterscheidung nennen wir die hier iterierte Funktion «normierte» Parabel, weil nur noch das quadratische Glied in der Formel auftritt.

Durch die Transformation

$$T: y \mapsto \frac{\mu}{a}y + \frac{1}{2}, [-1,1] \rightarrow [0,1]$$

Wobei $a \in]2,4]$ und $\mu = \frac{a(a-2)}{4}$, lässt sich diese Parabel auf einen Bereich des logistischen Wachstums zurückführen. Es gilt für $y \in [-1,1]$:

$$f(T(y)) = f\left(\frac{\mu}{a}y + \frac{1}{2}\right) = a\left(\frac{\mu}{a}y + \frac{1}{2}\right) - a\left(\frac{\mu}{a}y + \frac{1}{2}\right)^2 = \frac{a}{4} - \frac{\mu^2}{a}y^2$$

Und

$$T(g(y)) = T(1 - \mu y^2) = \frac{\mu}{a}(1 - \mu y^2) + \frac{1}{2} = \frac{\mu}{a} + \frac{1}{2} - \frac{\mu^2}{a}y^2 = \frac{a}{4} - \frac{\mu^2}{a}y^2$$

Denn

$$\frac{\mu}{a} + \frac{1}{2} = \frac{a-2}{4} + \frac{1}{2} = \frac{a}{4}$$

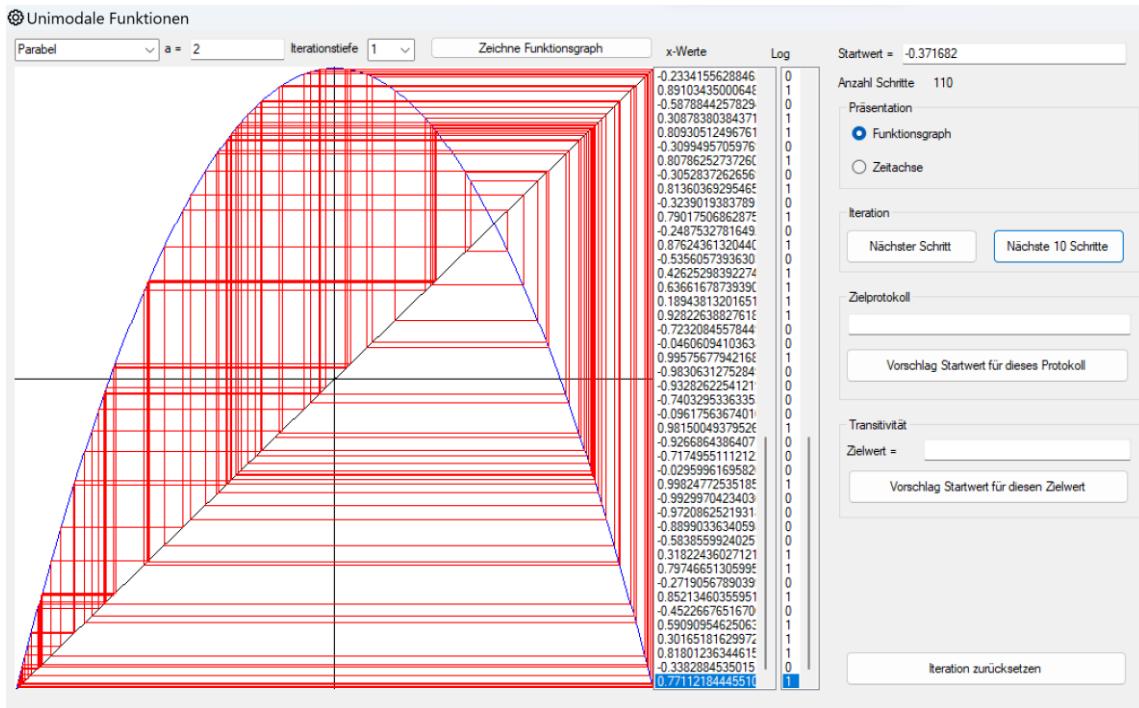
Somit ist $Tg \equiv fT$ bzw. $g \equiv T^{-1}fT$ auf $[-1,1]$. Für $a \in]2,4]$ ist $\mu \in]0,2]$.

Auch diese Parabel ist für $\mu = 2$ eine Konjugierte der Zeltabbildung. Die Transformation ist gegeben durch:

$$T: [0,1] \rightarrow [-1,1], u \in [0,1] \mapsto T(u) = \cos\pi(1-u) \in [-1,1]$$

T ist bijektiv, stetig differenzierbar und streng monoton wachsend. $T'(u) = \frac{1}{\pi} \sin\pi u > 0$ für $u \in [0,1]$.

Somit ist auch diese Funktion chaotisch für $a = 2$ (im «Simulator» wird der Parameter statt mit μ , mit a bezeichnet).



Iteration der normierten Parabel für $a = 2$

Wie das logistische Wachstum ist auch diese Parabel eine Konjugierte der Zeltabbildung, und zwar für den Parameterwert $\mu = 2$. Die Konjugation ist durch die Transformation gegeben:

$$T: [0,1] \rightarrow [-1,1], u \in [0,1] \mapsto x = T(u) = \cos(\pi(1-u)), x \in [-1,1]$$

Es gilt nämlich für $, u \in [0,1]$:

$$\begin{aligned} f(T(u)) &= f(\cos(\pi(1-u))) = 1 - 2\cos^2(\pi(1-u)) = \sin^2(\pi(1-u)) - \cos^2(\pi(1-u)) = \\ &= -\cos(2\pi(1-u)) = -\cos(2\pi u) = \cos(\pi(1-2u)) \end{aligned}$$

Und andererseits für die Zeltabbildung z :

$$T(z(u)) = \begin{cases} T(2u), u \in [0,0.5[\\ T(2(1-u)), u \in [0.5,1] \end{cases}$$

Im ersten Fall ist:

$$T(2u) = \cos(\pi(1-2u))$$

Im zweiten Fall ist:

$$T(2(1-u)) = \cos(\pi(1-2+2u)) = \cos(-\pi + 2\pi u) = \cos(\pi(1-2u))$$

Für $u \in [0,1]$ gilt also $fT \equiv Tz$ bzw. $f \equiv TzT^{-1}$.

Somit hat die von uns betrachtete Parabel für $\mu = 2$ chaotische Eigenschaften.

Protokolle

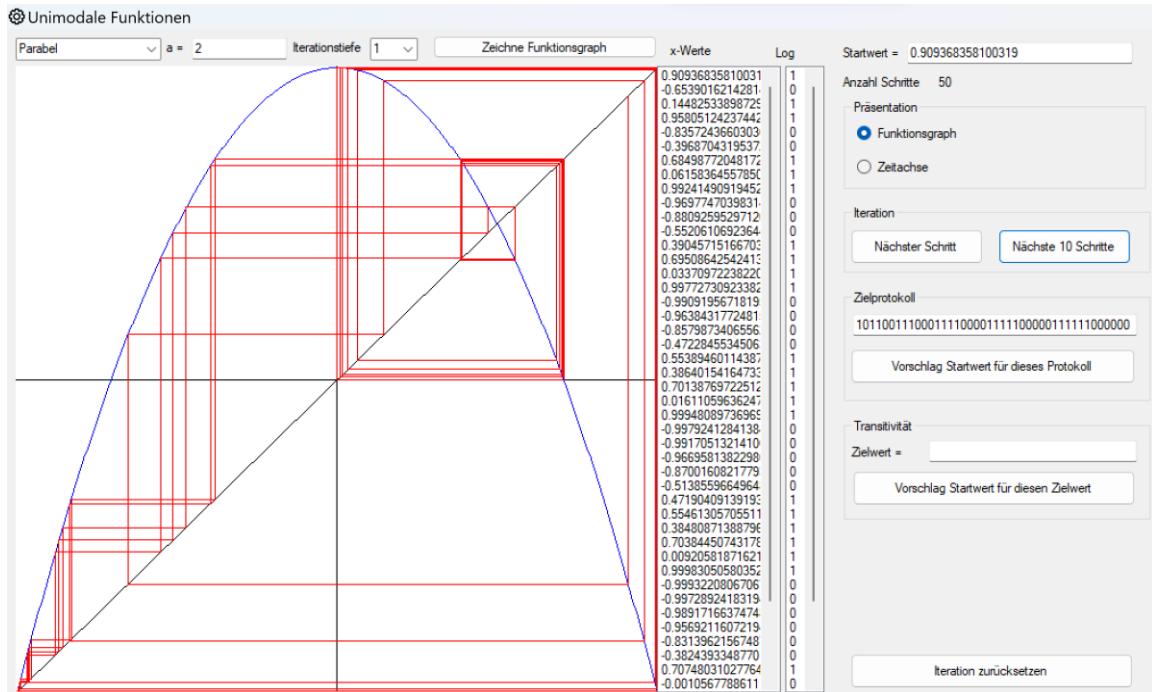
Wir definieren ein Protokoll wie folgt:

$$p(x_n) = \begin{cases} 0, & \text{falls } x_n \in [-1,0] \\ 1, & \text{falls } x_n \in]0,1] \end{cases}$$

Im chaotischen Fall $\mu = 2$ ist die hier betrachtete Parabel eine Konjugierte der Zeltabbildung. Für die Transformation gilt dann: $T(0.5) = -\cos \pi/2 = 0$ und T monoton wachsend. Somit gilt:

Die Protokolle der normierten Parabel für $\mu = 2$ und der Zeltabbildung stimmen bei der Iteration überein.

Das ermöglicht, wie beim logistischen Wachstum für ein vorgegebenes Protokoll zuerst einen entsprechenden Startwert für die Zeltabbildung zu finden, welcher dieses Protokoll liefert. Durch die Transformation T erhält man dann den entsprechenden Startwert für die normierte Parabel.



Iteration der normierten Parabel mit vorgegebenem Protokoll der Länge 42

3.6. Implementierung im «Simulator»

Für die Darstellung der Iteration steht das Fenster *FrmlIteration* zur Verfügung. Es ermöglicht die Wahl der iterierten Funktion, die Wahl des Parameters, die Festlegung der Iterationstiefe und die Wahl des Startwertes der Iteration. Seine Funktion ist im «Handbuch» detailliert beschrieben.

Das Resultat der Iteration kann im Funktionsgraph oder auf der Zeitachse dargestellt werden. Zu einem vorgegebenen Protokoll kann ein Startwert vorgeschlagen werden, welcher das vorgegebene

Protokoll liefert. Wenn im Falle der Transitivität ein Zielwert vorgegeben wird, dann wird ein neuer Startwert in der Nähe des ursprünglichen berechnet, so dass die Iterationsfolge dem Zielwert möglichst nahekommt.

Intern arbeitet dieses Fenster mit einem Interface *Iteration*, welches alle nötigen Eigenschaften verwaltet und Methoden anbietet. Dazu gehören:

- Interne Definition des zulässigen Parameterintervalls der Iteration und Test, ob ein vom User vorgegebener Parameterwert in diesem Intervall liegt
- Interne Definition des zulässigen Iterationsintervalls und Test, ob ein vom User vorgegebener Startwert in diesem Intervall liegt
- Interne Definition der Iterationsfunktion und der entsprechenden mehrfachiterierten Funktion, falls die Iterationstiefe > 1 ist.
- Eine Methode, welche zu einem vorgegebenen Protokoll einen entsprechenden Startwert berechnet
- Eine Methode, welche zu einem gegebenen Zielwert den Startwert so abändert, dass der Zielwert im Laufe der Iteration möglichst genau erreicht wird
- Durchführung der Iterationsschritte

Die *FrmIteration* übernimmt dann die Darstellung des Orbits der Iteration.

Die einzelnen Iterationsfunktionen

- Zeltabbildung
- Logistisches Wachstum
- «Normierte» Parabel

werden durch entsprechende Klassen dargestellt, welche das Interface *Iteration* implementieren. Dadurch ist es sehr einfach, weitere Beispiele von Iterationsfunktionen in das Programm einzufügen. Man kann einfach eine neue Klasse erstellen, welche das Interface *Iteration* implementiert.

Damit auch die Bezeichnungen klar sind, werden die Variablen für das logistische Wachstum und die normierte Parabel durchgehend mit (x, y) bezeichnet. Die Variablen für die Zeltabbildung dagegen mit (u, v) . Das entspricht auch den Bezeichnungen in dieser mathematischen Dokumentation.

Bei sämtlichen Berechnungen arbeiten wir in einem Koordinatensystem, welches durch einen Wertebereich für die x-Koordinate und y-Koordinate definiert ist.

$$x \in [x_{min}, x_{max}], y \in [y_{min}, y_{max}]$$

Ein Punkt (x, y) in diesem Koordinatensystem wird durch ein Objekt der Klasse *ClsMathPoint* repräsentiert.

Der Klasse *ClsGraphicTool* werden im Konstruktor die Picture Box bzw. die Bitmap übergeben, in welche gezeichnet werden soll, sowie die Intervalle $[x_{min}, x_{max}], [y_{min}, y_{max}]$. Die Klasse *ClsGraphicTool* besitzt dann die nötigen Methoden, um Zeichnungen zu generieren: *DrawCoordinatesystem*, *DrawPoint*, *DrawLine*, *DrawRectangle*, *DrawEllipse*, usw. Übergeben werden dabei immer die mathematischen Koordinaten bzw. Objekte vom Typ *ClsMathPoint*. Die Klasse *ClsGraphicTool* rechnet dann die mathematischen Koordinaten auf Grund der Eigenschaften der übergebenen Picture Box bzw. Bitmap in Pixel Koordinaten um. *Damit muss sich der Programmierer nicht um Pixelkoordinaten kümmern, sondern kann in mathematischen Koordinaten arbeiten.* Das gilt nicht nur im Falle der hier vorliegenden unimodalen Funktionen, sondern auch später beim Billard oder anderen Beispielen.

Um einen Startwert für ein vorgegebenes Protokoll oder einen vorgegebenen Zielwert zu finden, werden die entsprechenden Konjugations-Transformationen zur Zeltabbildung benutzt: Man berechnet zuerst den entsprechenden Startwert für die Zeltabbildung wie im entsprechenden vorhergehenden Abschnitt über die Zeltabbildung beschrieben. Dann erhält man den Startwert für das logistische Wachstum bzw. die normierte Parabel durch Anwendung der entsprechenden Konjugationstransformation, welche in den Abschnitten über diese Iterationen beschrieben sind.

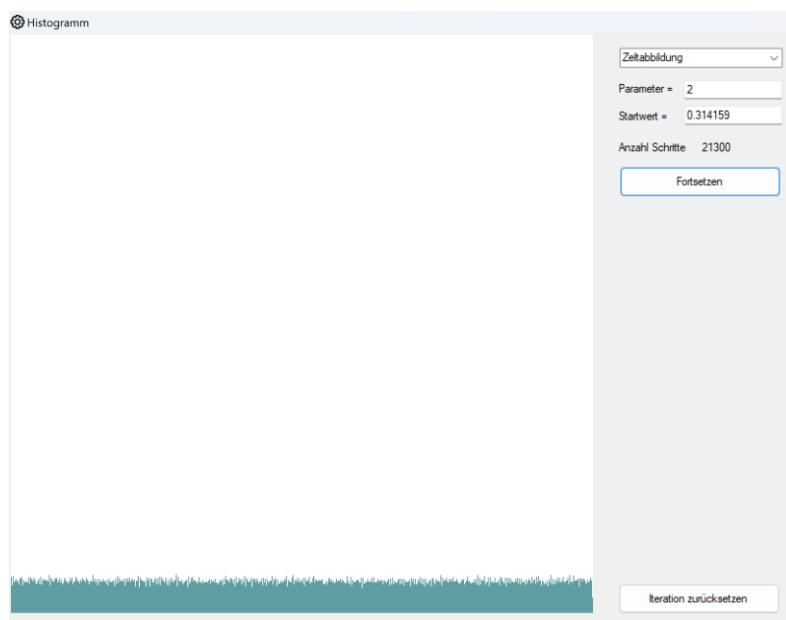
Bei der Darstellung der Sensitivität wird analog gearbeitet. Die *FrmSensitivity* ermöglicht die Wahl der entsprechenden Parameter und Einstellungen. Ihre Funktion ist im «Handbuch» detailliert beschrieben. Auch dort wird das Interface *Iteration* verwendet, welches durch die entsprechenden Klassen implementiert wird.

Wieder werden durchgehend dieselben Variablenbezeichnungen verwendet wie bei der *FrmIteration*.

Der Code ist überall mit detaillierten Kommentaren (auf Englisch) versehen. Das «Handbuch» bietet dem User einen Schnelleinstieg zur Benützung des «Simulator» an.

3.7. Histogramme im chaotischen Fall

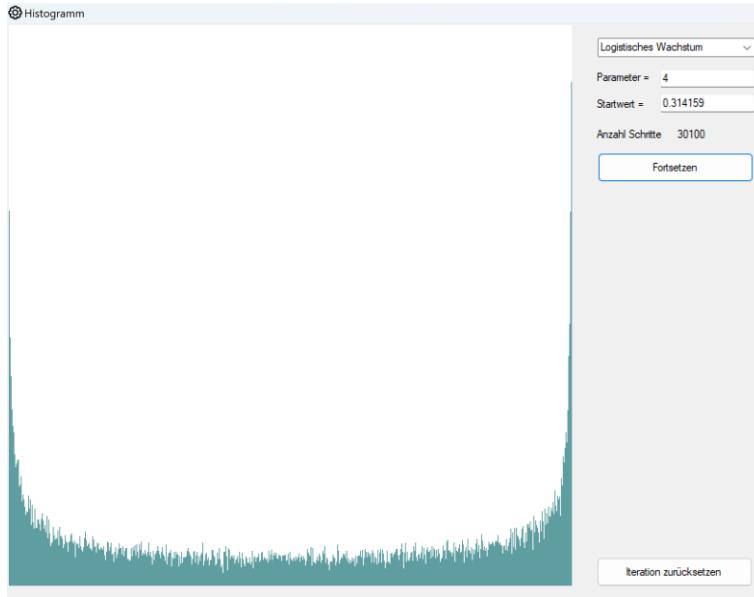
Hier betrachten wir den chaotischen Fall, also die Zeltabbildung, das logistische Wachstum mit Parameter $a = 4$ oder die Parabel mit Parameter $a = 2$. Es können auch andere Parameterwerte gewählt werden, aber dann erscheint der Hinweis, dass man dann die Eigenschaften der Sensitivität nicht garantiert hat. Ausgehend von einem (aperiodischen) Startwert $x_0 \in]0,1[$ für die Zeltabbildung oder das logistische Wachstum, bzw. $x_0 \in [-1,1]$ für die Parabel untersuchen wir, wie sich die Folge $f^n(x_0), n \in \mathbb{N}$ im Iterationsintervall «verteilt». Im «Simulator» unterteilen wir dieses Intervall in kleine Teilintervalle der Pixelbreite 1 und erstellen ein Histogramm, wie oft das Teilintervall im Laufe der Iteration getroffen wird.



Verteilung der getroffenen Intervalle bei der Zeltabbildung

Wie man sieht, entsteht bei der Zeltabbildung eine Gleichverteilung. Das heisst, dass die Iteration einen Orbit liefert, der zufällig aussieht, obwohl hinter der Iteration ein einfaches Bewegungsgesetz steckt. Man nennt diesen Effekt auch «Pseudozufall».

Bei der logistischen Abbildung entsteht ein ähnliches Bild, wobei die Randpunkte des Iterationsintervalls offenbar häufiger getroffen werden.



Histogramm des logistischen Wachstums nach 30'100 Schritten

Ausgehend von der gleichverteilten Zeltabbildung ist diese Verteilung ein Effekt der Transformation zum logistischen Wachstum gegeben durch $x = T(u) = \sin^2 \frac{\pi}{2} u$:

Wenn $p(x)$ die Wahrscheinlichkeitsverteilung für das logistische Wachstum ist, dann ist die Wahrscheinlichkeit, dass ein Iterationswert in ein Intervall Δx fällt, ungefähr $p(x)\Delta x$. Das ist aber gleich der Wahrscheinlichkeit, dass $u = T^{-1}(x)$ in das entsprechende Intervall Δu fällt. Diese Wahrscheinlichkeit ist aber gerade Δu , weil wir bei der Zeltabbildung auf dem Intervall $[0,1]$ eine Gleichverteilung haben. Es ist also $p(x)\Delta x \approx \Delta u$. Im Grenzwert gilt:

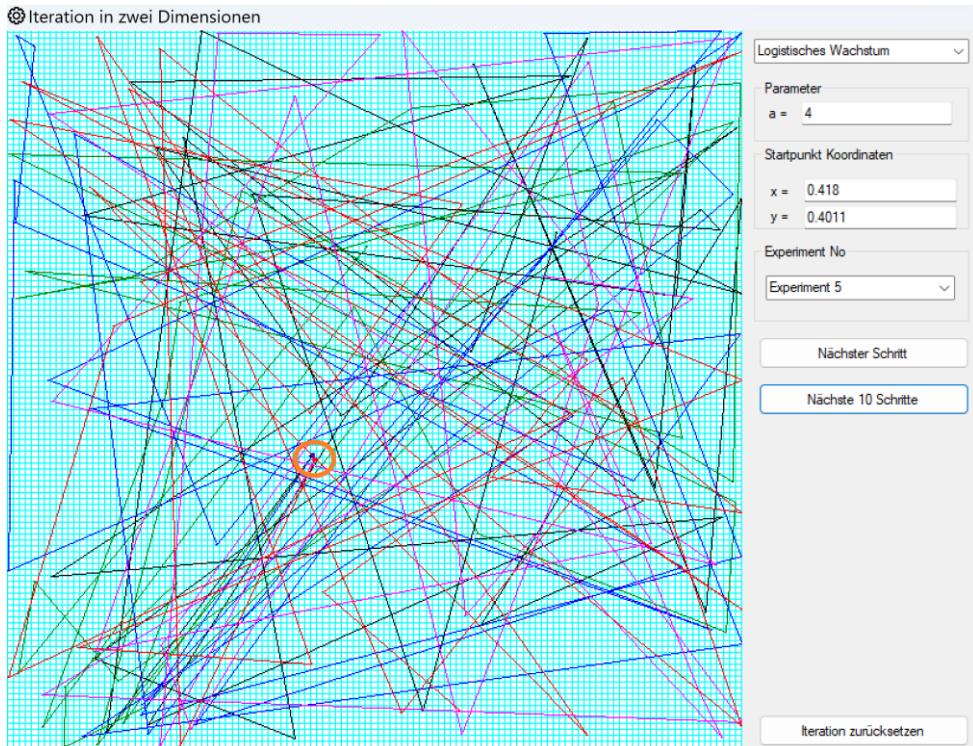
$$p(x) = \lim_{\Delta x \rightarrow 0} \frac{\Delta u}{\Delta x} = T^{-1}'(u) = \frac{1}{T'(u)} = \frac{2}{\pi \sin(\pi u)}$$

Das erklärt die Symmetrie des Histogramms des logistischen Wachstums bezüglich $x = 0.5$ bzw. $u = 0.5$ und auch die Form der Verteilung am Rande des Iterationsintervalls $[0,1]$.

3.8. Zweidimensionale Darstellung und Transitivität im chaotischen Fall

Im «Simulator» veranschaulichen wir die Situation eines Experimentators, welcher ein chaotisches System untersucht, hinter welchem das logistische Wachstum als Bewegungsgesetz steckt. Das System bewegt sich im Zustandsraum $[0,1] \times [0,1]$ und ein Systemzustand wird durch zwei Parameter $(x, y) \in [0,1] \times [0,1]$ gegeben. Der Experimentator wählt einen Startpunkt $(x_0, y_0) \in [0,1] \times [0,1]$ und untersucht dann, wie sich das System ausgehend von diesem Startpunkt verhält. Das Bewegungsgesetz lautet dann z.B. für das logistische Wachstum $x_{n+1} = 4x_n(1 - x_n)$ und $y_{n+1} = 4y_n(1 - y_n)$. Es ist also für jede Komponente dieselbe Iteration.

Der Experimentator misst mit einer begrenzten Genauigkeit. Im «Simulator» wird das so dargestellt, dass der Zustandsraum in sehr kleine Quadrate von 5x5 Pixel unterteilt wird, im Diagramm dargestellt durch ein Gitter. Das entspricht für einen x-Wert im Zentrum eines Quadrates, dass noch x-Werte mit einer Abweichung von ± 0.004 im selben Quadrat liegen. Dasselbe gilt analog für die y-Werte. Alle Startwerte in einem solchen Quadrat werden vom Experimentator als «derselbe» Startwert wahrgenommen. Dieser ist im folgenden Diagramm durch einen orangen Kreis markiert.



Zustandsraum des Experimentators im «Simulator»

Wenn sich nun die Startwerte in verschiedenen Experimenten nur so geringfügig unterscheiden, dass sie im selben Startquadrat zu liegen kommen, dann werden nach vielen Schritten wegen der Transitivität irgendwelche Zielwerte erreicht. Im obigen Bild führt der Experimentator 5 Experimente durch ausgehend vom Startwert, welcher mit einem kleinen Kreis markiert wird. In Wirklichkeit ist jeder Startwert minim verschoben. Für jedes Experiment wird der Orbit des Startpunktes in einer anderen Farbe dargestellt. Wie man sieht, sind diese Orbits nach wenigen Schritten deutlich unterschiedlich.

Für den Experimentator sieht das so aus, dass ausgehend vom selben Startwert irgendwelche Orbits generiert werden. Das System scheint sich zufällig zu verhalten und ein Bewegungsgesetz ist nicht ersichtlich.

3.9. Unimodale Funktionen

Bei Funktionen, bei denen der Zustandsraum $X = [a, b] \subset \mathbb{R}$ gestreckt und dann wieder gefaltet wird, lassen sich unter gewissen Bedingungen chaotische Eigenschaften feststellen. Zu den einfachsten solcher Abbildungen gehören die *unimodalen*. Alle bisherigen Beispiele von Iterationsfunktionen sind unimodal.

Eine stetige Abbildung $f: [a, b] \rightarrow [a, b]$, $[a, b] \subset \mathbb{R}$ heisst *unimodal*, falls ein Punkt $c \in [a, b]$ existiert, so dass gilt: f ist in $[a, c[$ streng monoton wachsend und in $]c, b]$ streng monoton fallend. Ferner soll gelten:

$$f(c) = b \text{ und } f(b) = a$$

Die Funktion f nimmt also in c ein Maximum an und das ist das einzige Maximum der Funktion. c heisst *kritischer Punkt* der Abbildung f .

Die Bedingung $f(b) = a$ wird oft für die Unimodalität nicht verlangt, bildet jedoch keine wesentliche Einschränkung.

Beispiele für unimodale Funktionen:

- Die Zeltabbildung. Hier ist $c = 0.5$
- Das logistische Wachstum für Parameter $a \in]0,4]$. Dann ist $c = 0.5$
- Die normierte Parabel für alle Parameterwerte $a \in]0,2]$. Hier ist $c = 0$

Unimodale Funktionen sind eine Verallgemeinerung der Zeltabbildung und ihrer Konjugierten.

Eine allgemeine Diskussion dieser Funktionen würde in diesem Dokument zu weit führen. Wir erwähnen lediglich einen Satz und verweisen auf die Literatur, siehe [5].

Satz

Sei eine Funktion $f: [a, b] \rightarrow [a, b] \subset \mathbb{R}$ unimodal und gegeben durch ein Polynom zweiten Grades, dann gilt:

- Es gibt höchstens einen attraktiven Zyklus in $[a, b]$
- Falls es einen solchen attraktiven Zyklus gibt, dann liegt der kritische Punkt in seinem Bassin

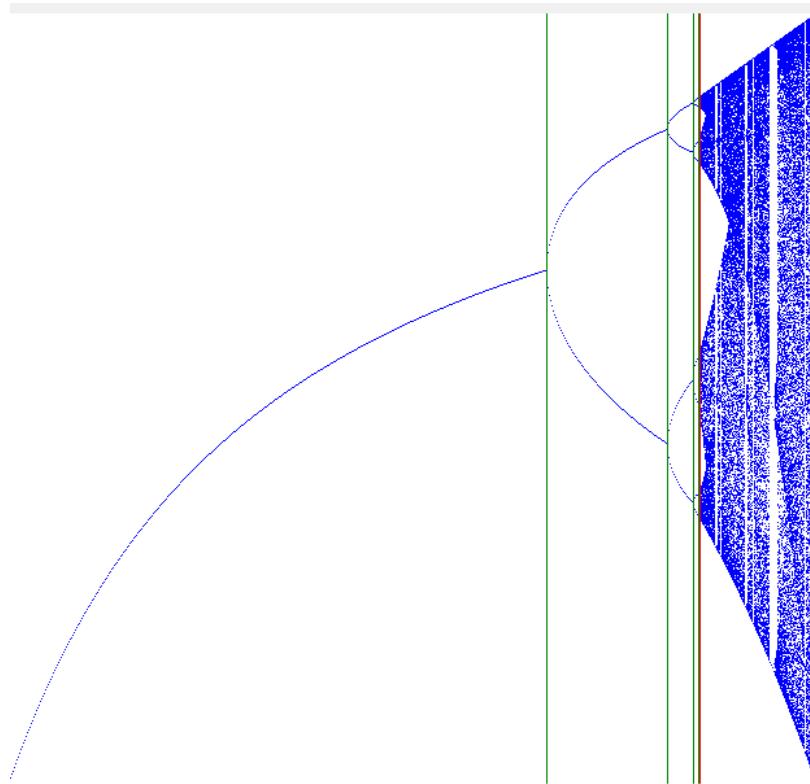
Für die Untersuchung chaotischer Eigenschaften genügt es bereits, unimodale Funktionen zu untersuchen, welche durch quadratische Polynome definiert sind. Nun wissen wir auch, dass es genügt, den Orbit des kritischen Punktes zu untersuchen, also der Stelle, in welche die quadratische Funktion das Maximum annimmt.

3.10. Periodenverdoppelung

Wir haben gesehen, dass das logistische Wachstum für Parameterwerte $a < 1$ den attraktiven Fixpunkt $\xi_1 = 0$ hat. Für $a \in]1, 3[$ wird dieser Fixpunkt repulsiv, aber an seine Stelle tritt der attraktive Fixpunkt $\xi_2 = 1 - 1/a$. Dieser ist attraktiv für $a \in]1, 3[$. An der Stelle $a = 3$ wird dieser Fixpunkt repulsiv, aber es entsteht ein attraktiver 2-er Zyklus

$$\xi_{3,4} = \begin{cases} 0.799455 \dots \\ 0.513044 \dots \end{cases}$$

Dieser ist attraktiv im Bereich $a \in]3, 1 + \sqrt{6}[$ und wird an der Stelle $a = 1 + \sqrt{6}$ repulsiv. Das Computerexperiment zeigt dann, dass hier ein attraktiver 4-er Zyklus entsteht. Für wachsende a entsteht dann ein 8-Zyklus, dann ein 16-Zyklus und die Periode der Zyklen verdoppelt sich laufend bis zu einem gewissen Grenzwert von a und geht dann über zu chaotischem Verhalten. Das wird in folgendem Diagramm gezeigt, welches nach dem Mathematiker Mitchell Feigenbaum benannt ist, der 1975 das Phänomen der Periodenverdoppelung als vertieft untersucht hat.



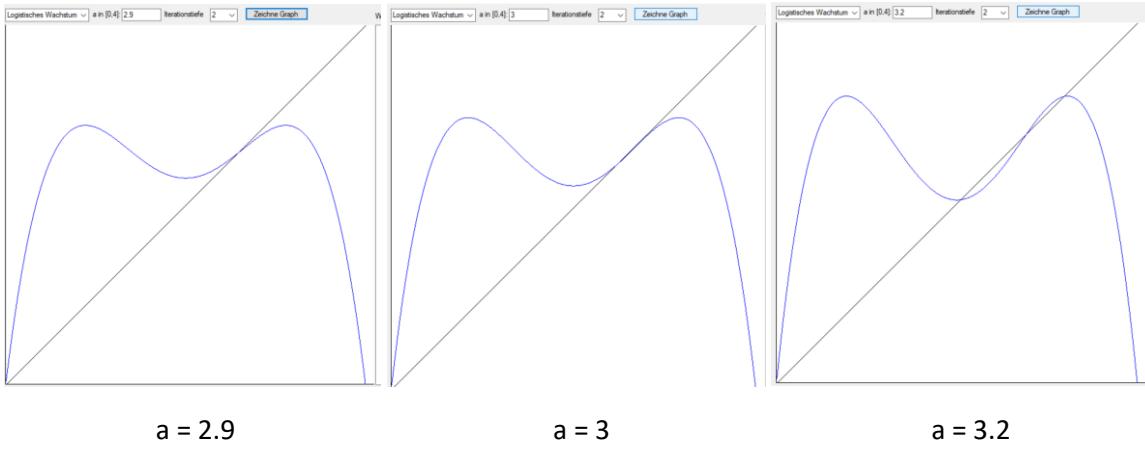
Feigenbaum-Diagramm für das logistische Wachstum

Auf der horizontalen Achse sind die Werte des Parameters a dargestellt. Das Intervall, in welchem sich a bewegt, kann innerhalb $]0,4]$ frei gewählt werden. Im obigen Bild sieht man einen Ausschnitt für $a \in [1, 4]$. Für jeden Wert von a wird dann so lange iteriert, dass man hoffen kann, dass sich die Iteration entweder auf einen Zyklus einpendelt oder chaotisch ist. Anschliessend weitere Iterationsschritte durchgeführt und die x -Werte der Iteration in vertikaler Richtung ins Diagramm eingetragen.

Die grünen Linien zeigen die Split Punkte auf, an denen ein attraktiver Zyklus instabil wird und ein neuer Zyklus entsteht. Den ersten Split Punkt hat man für $a_1 = 3$. Dort entsteht ein attraktiver 2-Zyklus. An der Stelle $a_2 = 3.449499 \dots$ verdoppelt sich die Periode und es entsteht ein attraktiver 4-Zyklus bei den entsprechenden zwei Split Punkten. An der Stelle $a_3 = 3.544090 \dots$ entsteht bei vier Split Punkten der attraktive 8-Zyklus. Im weiteren Verlauf verdoppelt sich die Periode in immer kürzeren Abständen und die Folge der a_i , also den Stellen an denen weitere Split Punkte auftreten, strebt gegen den Grenzwert $a_\infty = 3.569946 \dots$. Diese Stelle ist im Diagramm mit einer roten Linie markiert.

Die Split Punkte gehören jeweils zu einem 2^n -Zyklus, welcher an dieser Stelle attraktiv wird. Da alle zu einem Zyklus gehörenden Punkte denselben Multiplikator haben, wie bereits früher festgestellt, werden sie gleichzeitig attraktiv, das heisst, sie liegen alle auf einer Geraden parallel zur vertikalen Achse.

Was zum Beispiel beim Übergang des Fixpunktes $\xi_2 = 1 - 1/a$ in einen attraktiven 2-Zyklus an der Stelle $a=3$ passiert, sieht man beim Graph von f^2 an dieser Stelle (generiert vom «Simulator»):



Der Fixpunkt ξ_2 ist der Schnittpunkt des Graphen mit der 45° -Geraden. Links hat die Kurve eine Tangente mit Steigung < 1 , der Fixpunkt ist also attraktiv. Für $a = 3$ ist die Tangentensteigung genau 1 (mittleres Bild). Wenn für $a = 3.2$ die Kurve steiler wird, ist der Fixpunkt ξ_2 repulsiv geworden mit einer Tangentensteigung > 1 . Dafür sind zwei neue Schnittpunkte entstanden mit Tangentensteigung < 1 . Das ist der neu entstandene 2-Zyklus.

Die ersten zugehörigen Werte von a , an denen die jeweiligen Zyklen instabil werden und sich in einen Zyklus der doppelten Periode aufsplitten, können mit numerischen Methoden ermittelt werden und sind:

$$a_1 = 3, a_2 = 3.449499 \dots, a_3 = 3.544090 \dots, a_4 = 3.564407 \dots,$$

$$a_5 = 3.568759 \dots, a_6 = 3.569692 \dots, a_7 = 3.569891 \dots, a_8 = 3.569934 \dots$$

Die Werte von $a_k, k \in \mathbb{N}$ nehmen geometrisch ab und streben gegen einen Grenzwert a_∞ gemäss:

$$a_k \approx a_\infty - c \cdot \delta^{-k}, k \in \mathbb{N}$$

$\delta \approx 4.669202 \dots$ ist die sogenannte Feigenbaum Konstante. Für das logistische Wachstum gilt:

$$c \approx 2.6327 \dots$$

Zum Beispiel ist:

$$a_\infty - c \cdot \delta^{-5} = 3.568759 \approx a_5$$

Ein n -Zyklus erfüllt die Bedingung $f^n(\xi_i) = \xi_i, \forall \xi_i \in \text{Zyklus}, 1 \leq i \leq 2$. Beim logistischen Wachstum sind diese Punkte also Nullstellen des Polynoms $f^n(x) - x = 0$, welches den Grad 2^n hat. Kann es sein, dass einige dieser Nullstellen komplex und die anderen reell sind? Die Antwort ist nein. Angenommen, man kennt eine reelle Nullstelle ξ_1 . Dann ergeben sich die weiteren Nullstellen aus der Iteration $\xi_{i+1} = f(\xi_i), 1 \leq i < n$ und $\xi_1 = f(\xi_n)$. Das heisst, alle weiteren Nullstellen sind ebenfalls reell.

Feigenbaum hat entdeckt, dass das Verhalten der Periodenverdoppelung ein universelles Phänomen ist und in vielen dynamischen Systemen beim Übergang zum Chaos auftritt. Dabei ist die Konstante \mathcal{F} immer dieselbe und scheint ebenfalls universell.

Im Falle der Parabel erhält man die Split Punkte durch Anwendung der Transformation zwischen logistischem Wachstum und Parabel: $\mu = \frac{a(a-2)}{4}$

Das liefert für die Parabel folgende Stellen, an denen Split Punkte auftreten:

$$\mu_1 \approx 0.75, \mu_2 \approx 1.24995, \mu_3 \approx 1.3681, \mu_4 \approx 1.39405, \dots, \mu_\infty \approx 1.401155$$

Auch hier folgt die Folge $(\mu)_k$ demselben Gesetz. Die zugehörige Konstante \tilde{c} bestimmen wir mit Hilfe der Transformation zwischen logistischem Wachstum und der Parabel wie folgt:

$$\begin{aligned}\tilde{c} &= \lim_{k \rightarrow \infty} (\mu_\infty - \mu_k) \delta^k = \lim_{k \rightarrow \infty} \frac{\delta^k}{4} (a_\infty(a_\infty - 2) - a_k(a_k - 2)) \\ &= \lim_{k \rightarrow \infty} \frac{\delta^k}{4} ((a_\infty - a_k)(a_\infty + a_k) - 2(a_\infty - a_k)) = \lim_{k \rightarrow \infty} \frac{\delta^k}{4} (a_\infty - a_k)(a_\infty + a_k - 2) \\ &= \lim_{k \rightarrow \infty} \frac{\delta^k}{4} c \delta^{-k} (a_\infty + a_k - 2) = \frac{c(a_\infty - 1)}{2} = 3.3829484 \dots\end{aligned}$$

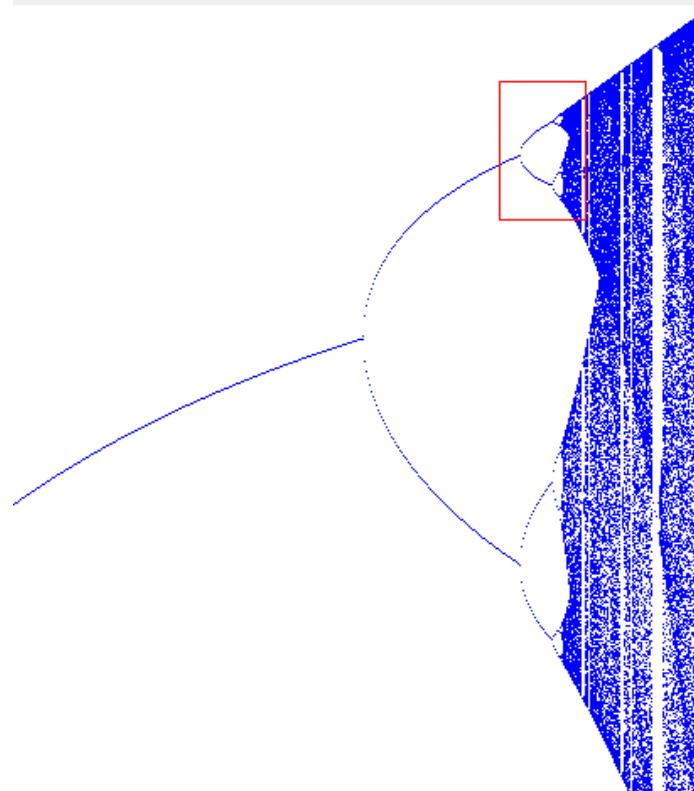
Es gilt dann zum Beispiel:

$$\mu_\infty - \tilde{c} \cdot \delta^{-4} = 1.3940375 \approx \mu_4$$

Dabei folgt die Skalierung in a-Richtung etwa der obigen geometrischen Folge mit der Feigenbaum Konstanten $\delta \approx 4.669202 \dots$. Auch die Skalierung in x-Richtung folgt einer geometrischen Folge, wobei auch hier ein universeller Skalierungsfaktor auftritt, nämlich: $\alpha \approx 2.5029078 \dots$

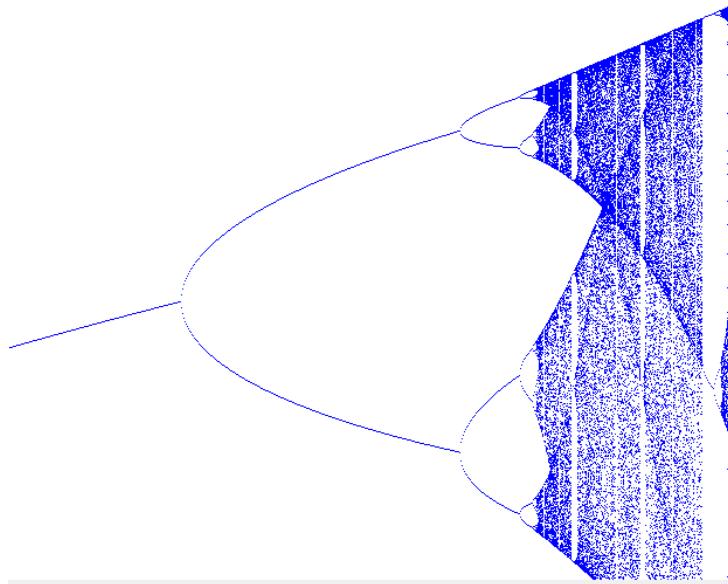
Selbstähnlichkeit und Renormalisation

Feigenbaum verwendete bei seiner Untersuchung das Phänomen der Selbstähnlichkeit einer quadratischen Funktion und deren Iterierten. Eine gute Darstellung über diesen Ansatz findet man in [1] und einen elementaren Zugang in [4].



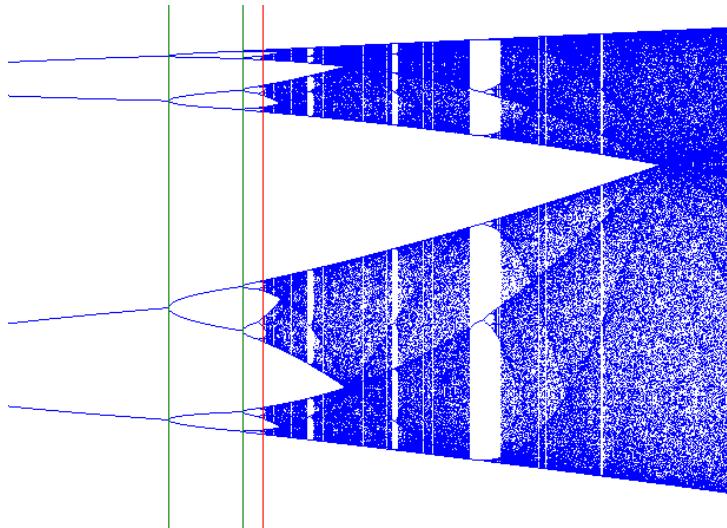
Selbstähnlichkeit bei der Periodenverdoppelung

Der «Simulator» ermöglicht es, den mit einem roten Rechteck markierten Ausschnitt in einer entsprechend skalierten Iteration zu betrachten. Man erhält für den obigen Ausschnitt folgendes Bild:



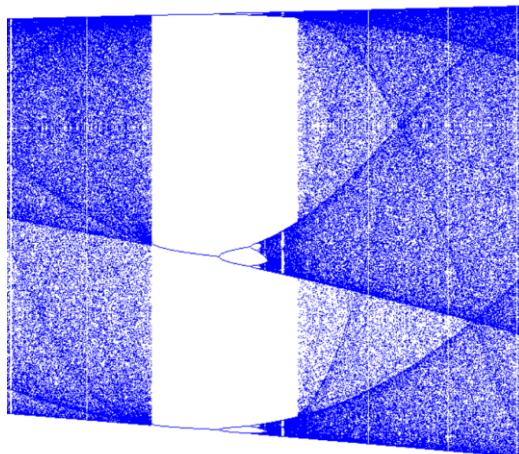
Rot markierter Ausschnitt in vergrösserter Form

Wenn a den Wert $a_\infty \approx 3.569946$ überschreitet, wird das System chaotisch.



Übergang ins Chaos – die rote Linie markiert ungefähr den Wert $a_\infty \approx 3.569946$

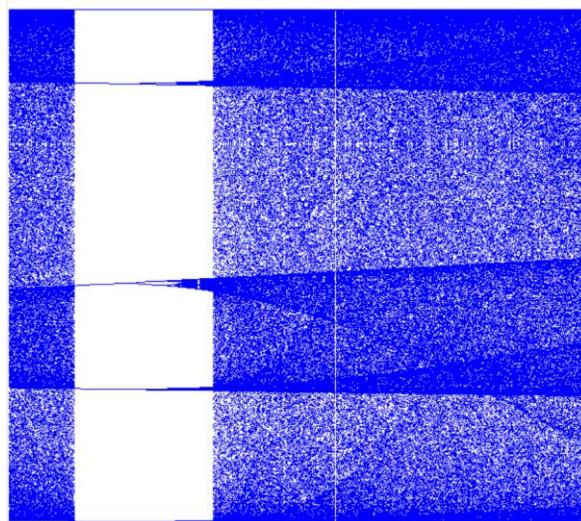
Wenn a weiter zunimmt, wechseln sich chaotische Bereich mit immer kleineren Fenstern ab, in denen sich attraktive Zyklen befinden. Im Bereich $a \in [3.828427, 3.841499]$ taucht z.B. ein attraktiver 3-Zyklus auf, der dann an der rechten Intervallgrenze instabil wird. An dieser Stelle findet wieder eine Kaskade von Periodenverdoppelungen statt: Aus dem 3-Zyklus wird ein 6-Zyklus, dann ein 12-Zyklus usw. Bei $a = 4$ wird das Chaos definitiv erreicht.



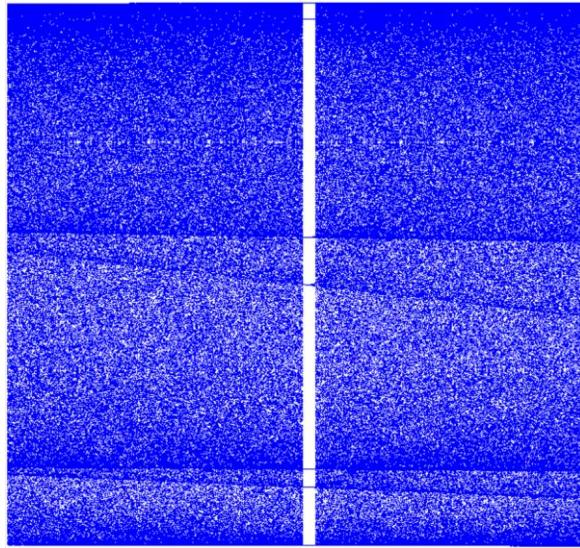
Attraktiver 3-Zyklus im Bereich $\alpha \in [3.8, 3.9]$, erzeugt vom «Simulator»

Man sieht im obigen Bild, wie etwa an der Stelle $\alpha=3.82$ nach einem chaotischen Bereich wie aus dem Nichts ein stabiler 3-Zyklus auftaucht, welcher dann in die Periodenverdoppelung übergeht, und später wieder im Chaos landet. Etwas rechts von der Periodenverdoppelung sieht man die Selbstähnlichkeit: Auch hier tauchen Fenster mit stabilen 9-Zyklen auf, ähnlich zum 3-Zyklus.

Auch im Bereich $\alpha \in [3.905, 3.91]$ tauchen noch Fenster mit stabilen Zyklen auf, wie das folgende Bild zeigt. Man erkennt z.B. auf der linken Seite des Diagramms einen stabilen 5-Zyklus. Bei genauem Hinsehen erkennt man schwach, dass er in die Periodenverdoppelung übergeht, sobald er instabil wird.



Attraktiver 5-Zyklus im Bereich $\alpha \in [3.905, 3.91]$



Attraktiver 7-Zyklus im Bereich $a \in [3.95, 3.952]$

In diesem Zusammenhang sei ein Satz von Sarkovskii erwähnt, der hier nicht bewiesen werden kann, aber einen Eindruck über die auftretenden Zyklen gibt:

Satz (Sarkovskii 1964)

Betrachte folgende Ordnung auf \mathbb{N} :

$$3 > 5 > 7 > 9 > \dots > 2 \cdot 3 > 2 \cdot 5 > 2 \cdot 7 > \dots > 2^n \cdot 3 > 2^n \cdot 5 > \dots > 2^n > \dots > 4 > 2 > 1$$

Sei ferner f eine unimodale Funktion, welche einen p -periodischen Zyklus hat. Dann besitzt f Zyklen jeder Periode $q < p$ im Sinne der obigen Ordnung.

Eine Begründung dieses Satzes übersteigt die Möglichkeiten dieses Manuskriptes. Man findet aber einen guten Zugang in [1]. Einen elementaren Zugang findet man in [6].

Beispiel

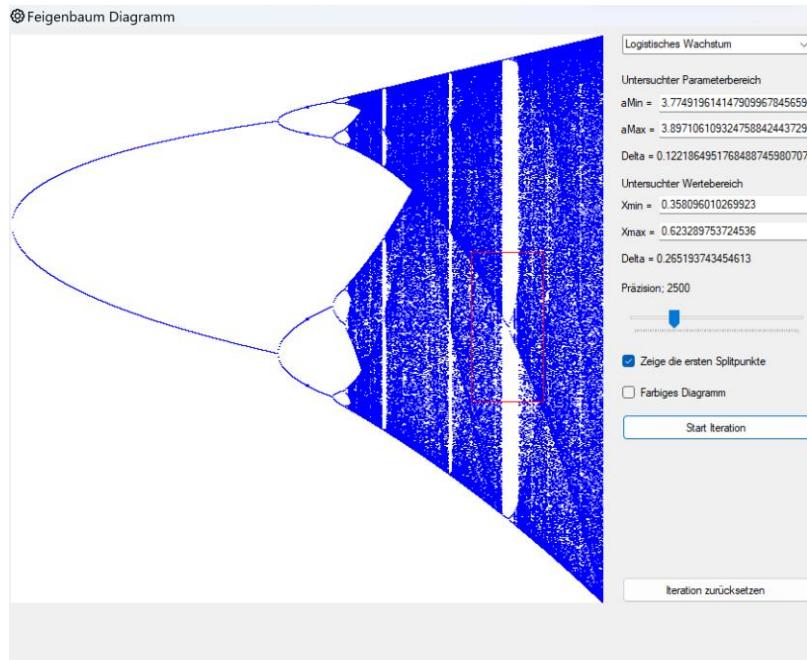
Das logistische Wachstum ist definiert durch eine unimodale Funktion. Die Computerexperimente haben einen attraktiven 3-Zyklus gezeigt. Somit besitzt das logistische Wachstum mindestens alle Zyklen der Periode $(2k + 1) \cdot 2^n, k \in \mathbb{N}, n \in \mathbb{N}_0$.

3.11. Implementierung im «Simulator»

Die *FrmFeigenbaum* unterstützt die Untersuchung der allfälligen Zyklen in Abhängigkeit eines Parameters a für die Iteration. Auch hier verwendet die *FrmFeigenbaum* die Methoden und Eigenschaften des Interfaces *IIterator*. Klassen, welche dieses Interface implementieren, enthalten dann die Spezifika der jeweiligen Iteration.

Der Benutzer kann den Parameterbereich, in welchem sich a bewegt, manuell eingeben. Ebenso kann er für die x-Werte einen Wertebereich eingeben, der festlegt, in welchem Bereich die x-Werte betrachtet werden. Das führt zu einer Skalierung der x-Achse.

Beides kann auch durch eine Auswahl mit gedrückter linker Maustaste geschehen. So hat man die Möglichkeit, interessante Bereich im Feigenbaum Diagramm näher zu untersuchen. Eine detaillierte Beschreibung findet man im «Handbuch».



Ausschnitt aus dem Feigenbaum Diagramm mit einer Auswahl des Benutzers

Die Daten des roten Auswahlrechtecks werden im «Simulator» angezeigt. Wenn man das rote Rechteck, welches den Ausschnitt des Diagrammes bestimmt, auf die Split Punkte a_i legt, sieht man damit die Differenz $\Delta a_i := a_{i+1} - a_i$. Man kann dann einen Näherungswert der Feigenbaum Konstante berechnen:

$$\delta \approx \frac{\Delta a_i}{\Delta a_{i+1}}$$

Ebenso sieht man die Differenz zweier aufeinanderfolgender Zykluspunkte: $\Delta x_i := \xi_{i+1} - \xi_i$ wenn man das Rechteck auf diese legt. Damit erhält man eine Näherung für den skalierenden Faktor in x-Richtung:

$$\alpha \approx \frac{\Delta x_i}{\Delta x_{i+1}}$$

3.12. Übungsbeispiele

1. Gegeben ist die Funktion:

$$f(x) = ax\sqrt{1-x^2}, \quad I \rightarrow I, a \in P$$

Wobei der Iterationsbereich I und der Parameterbereich P endliche reelle Intervalle sind.

- a) Wie ist I zu wählen, damit die Funktion unimodal ist und als Iterationsvorschrift verwendet werden kann?
- b) In welchem Bereich darf der Parameter a dann liegen?
- c) Bestimme Fixpunkte und 2-Zyklen der zugehörigen Iteration. Für welche Werte von a sind diese attraktiv bzw. repulsiv?
- d) Erweitere den «Simulator» mit einer Klasse *ClsRoot*, welche obige Iteration definiert und das Interface *Iterator* implementiert. Experimentiere mit dem «Simulator».

e) Zeige, dass die Iteration für ein gewisses a zur Zeltabbildung konjugiert ist (verwende trigonometrische Funktionen).

2. Untersuche die Sägezahnabbildung:

$$f(x) = \begin{cases} 2x, & x \in [0, 0.5[\\ 2x - 1, & x \in [0.5, 1] \end{cases}$$

Zeige mit Hilfe der Dualbruchdarstellung von x , dass

- a) die periodischen Punkte dicht im Intervall $[0, 1]$ liegen und dass sie alle repulsiv sind.
- b) die Abbildung sensitiv ist
- c) die Abbildung transitiv ist
- d) Wenn man ein Protokoll definiert: «0» falls $x \in [0, 0.5[$ und «1» falls $x \in [0.5, 1]$. Dann gibt es zu jedem vorgegebenen Protokoll ein Startwert, der dieses Protokoll liefert.
- e) Erweitere den «Simulator» mit einer Klasse `ClsSawTooth`, welche obige Iteration definiert und das Interface `IIteration` implementiert. Experimentiere mit dem «Simulator».
- f) Die Parabel ist für den Parameter $a = 2$ eine Konjugierte der Zeltabbildung und die zugehörige Transformation ist $T(u) = \cos(\pi(1 - u)), u \in [0, 1]$. Zeige: Die Parabel ist für $a = 2$ auch eine Konjugierte der Sägezahnabbildung und die zugehörige Transformation ist:

$$T(u) = \cos(\pi u), u \in [0, 1]$$

3. Untersuche die Winkelverdoppelung am Einheitskreis hinsichtlich chaotischem Verhalten.

Verwende dazu entweder die Abbildung $\varphi \rightarrow e^{i\varphi}$ in der komplexen Ebene oder $\varphi \rightarrow \begin{pmatrix} \cos \varphi \\ \sin \varphi \end{pmatrix}$ im \mathbb{R}^2 .

4. Untersuche die Funktion $f(x) = ax^2(1 - x^2), [0, 1] \rightarrow [0, 1]$.

- a) In welchem Intervall darf der Parameter a liegen?
- b) Bestimme Fixpunkte und 2-Zyklen
- c) für welche a sind diese attraktiv? Repulsiv?
- d) Erweitere den «Simulator» mit einer Klasse, welche das Interface `IIteration` implementiert, und diese Funktion darstellt.

5. Untersuche das Feigenbaum Diagramm mit dem «Simulator». Wähle das rote Auswahldreieck so, dass die Parameterdifferenzen aufeinanderfolgender Split Punkte getroffen werden. Dann werden rechts diese Differenzen angezeigt. Ermittle damit Näherungswerte für die Feigenbaumkonstante.

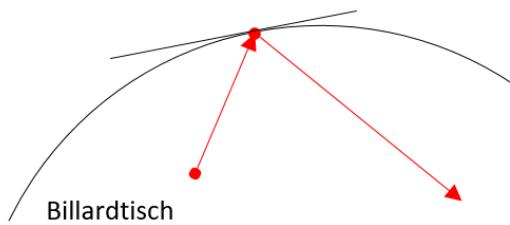
Weitere Übungsbeispiele findet man in [1] und [2].

4. Mathematisches Billard

Vorbemerkung: Dieses Kapitel ist unabhängig von den anderen und nimmt nur auf Kapitel zwei Bezug.

4.1. Einführung

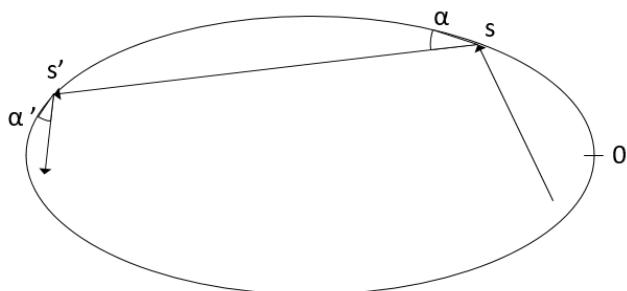
In diesem Papier betrachten wir als Billardtisch ein konvexes Gebiet der Ebene. Auf dem Billardtisch bewegt sich eine idealisierte (masselose und punktförmige) Billardkugel reibungsfrei und geradlinig, bis sie auf den Rand des Billardtisches trifft. Hier wird sie gemäss dem Reflexionsgesetz Ausfallwinkel = Einfallwinkel reflektiert. Der Einfallwinkel ist als Winkel zwischen der Kugelbahn vor dem Stoss und der Tangente an die Randkurve im Stosspunkt definiert. Ebenso ist der Ausfallwinkel der Winkel zwischen dieser Tangente im Stosspunkt und der Kugelbahn nach dem Stoss.



Reflexion der Billardkugel (rot) am Rand des Billardtisches

Speziell werden wir den elliptischen, den ovalen und den stadionförmigen Billardtisch untersuchen. Im Programm «Simulator» sind diese Tischformen implementiert und es können beliebig viele Kugeln gestartet werden. Über den Menüpunkt «Mechanik – Billard» kommt man zu diesen Funktionen.

Als Parametrisierung dieser Form des Billards wird oft auf dem Rand des Billardtisches ein Nullpunkt festgelegt. Die Lage eines Stosspunktes wird dann definiert durch die Bogenlänge s der Randkurve zwischen diesem Stosspunkt und dem Nullpunkt. Als zweiten Parameter kann man den Winkel α zwischen Kugelbahn und der Tangente im Stosspunkt angeben. Zusammen mit dem Reflexionsgesetz ist das Bewegungsgesetz für das Billard damit definiert.



Parametrisierung der Bahn einer Billardkugel

Es ist eine Abbildung, welche einem Stosspunkt s und einem Ausfallwinkel α den nächsten Stosspunkt s' und den nächsten Ausfallwinkel α' zuordnet:

$$(s, \alpha) \rightarrow (s', \alpha')$$

Dabei ist s eine positive reelle Zahl und α ein Winkel zwischen 0 und π .

Wir werden später einige allgemeine Überlegungen zum Billard machen. Zuerst wenden wir uns den konkreten Beispielen zu.

4.2. Grundlagen des elliptischen Billards

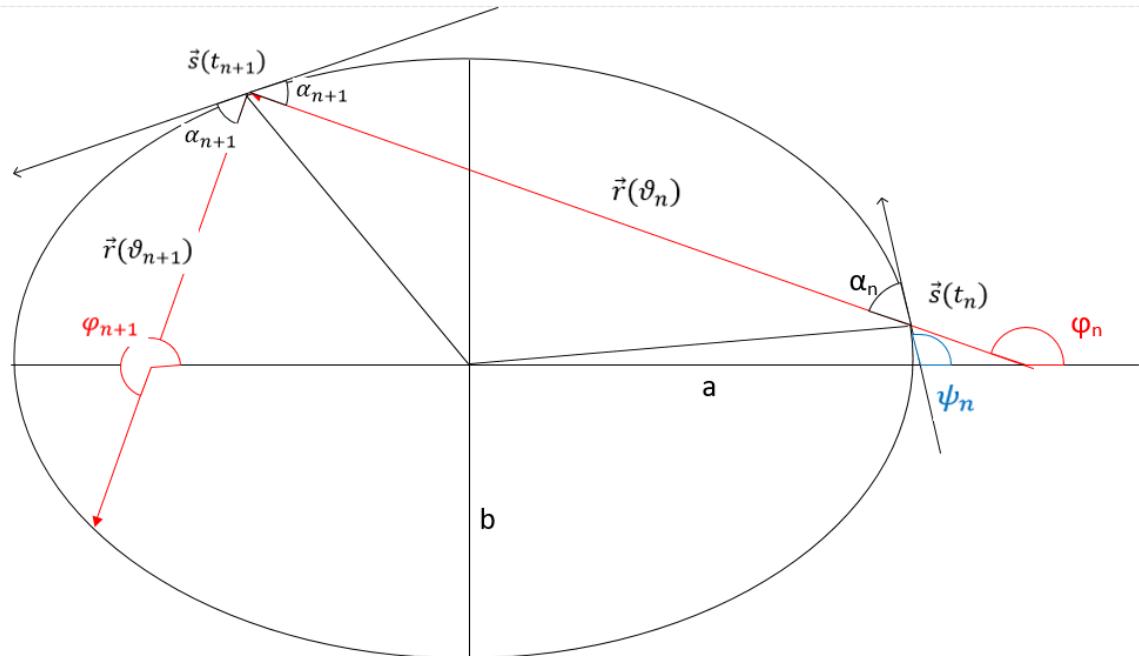
Im Falle eines elliptischen Billardtisches ist die Parametrisierung durch die Bogenlänge auf der Randkurve nicht für eine Programmierung geeignet. Wir verwenden an deren Stelle die übliche Parameterdarstellung der Ellipse.

Eine Ellipse mit Hauptachsen a und b wird parametrisiert durch:

$$\vec{s}(t) = \begin{pmatrix} a \cos t \\ b \sin t \end{pmatrix}, t \in [0, 2\pi[, a, b \in \mathbb{R}^+$$

Nun betrachten wir eine Kugel, welche reibungsfrei auf dem elliptischen Tisch rollt und vom Tischrand reibungsfrei reflektiert wird.

Bemerkung: Aus optischer Sicht könnte man an Stelle der Kugelbahn einen Lichtstrahl betrachten, welcher an einem elliptischen Spiegel reflektiert wird. Das liefert dasselbe Szenario.



Elliptisches Billard

Nach dem n -ten Stoß startet die Kugel in einem Punkt

$$\vec{s}(t_n) = \begin{pmatrix} a \cos t_n \\ b \sin t_n \end{pmatrix}$$

In eine Richtung, welche durch den Ausfallwinkel α_n definiert ist (siehe obige Skizze).

Sei ψ_n der Winkel zwischen Tangente im Stoßpunkt $\vec{s}(t_n)$ und der positiven x-Achse. Mit ϕ_n bezeichnen wir den Winkel der Kugelbahn nach dem n -ten Stoß und der positiven x-Achse. Dann gilt:

$$\varphi_n = \psi_n + \alpha_n$$

Die Kugel bewegt sich dann in eine Richtung gegeben durch:

$$\vec{r}(\vartheta_n) = \begin{pmatrix} a \cos \vartheta_n \\ b \sin \vartheta_n \end{pmatrix}, \vartheta_n \in [0, 2\pi[$$

Beachte: Der Parameter ϑ ist nicht identisch mit dem Winkel φ !

Da $\begin{pmatrix} a \cos \vartheta \\ b \sin \vartheta \end{pmatrix}$ parallel zu $\begin{pmatrix} \cos \varphi \\ \sin \varphi \end{pmatrix}$ sein soll, gilt:

$$(1): \vartheta = \begin{cases} \arctan\left(\frac{a}{b} \tan \varphi\right), \varphi \in [0, \pi/2[\\ \pi + \arctan\left(\frac{a}{b} \tan \varphi\right), \varphi \in]\pi/2, 3\pi/2[\\ 2\pi + \arctan\left(\frac{a}{b} \tan \varphi\right), \varphi \in]3\pi/2, 2\pi[\\ \varphi, \varphi \in \{\pi/2, 3\pi/2\} \end{cases}$$

Dabei verwendet man den Zweig der arctan Funktion entsprechend der im Programm verfügbaren arctan Funktion:

$$\arctan(x) : \mathbb{R} \rightarrow]-\pi/2, +\pi/2[$$

Im «Simulator» wird diese Umrechnung von φ nach ϑ in der Klasse *ClsMathHelperBillard* abgehandelt und das Intervall, in welchem sich φ bewegt, wird mit dem Vorzeichen von $\cos \varphi$ und $\sin \varphi$ ermittelt.

Die Kugel bewegt sich nun auf der Geraden g:

$$\vec{x}(u) = \vec{s}(t_n) + u \cdot \vec{r}(\vartheta_n), u \in \mathbb{R}$$

Beim nächsten Stoss trifft die Kugel auf den Punkt:

$$\vec{s}(t_{n+1}) = \begin{pmatrix} a \cos t_{n+1} \\ b \sin t_{n+1} \end{pmatrix}$$

Dieser Punkt ergibt sich als Schnitt zwischen der Geraden g und der Ellipse. Wir verwenden die Koordinatenform der Ellipsengleichung:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

Und setzen darin die Koordinaten eines Punktes auf der Geraden ein. Das ergibt als Bedingung für den Schnittpunkt:

$$\frac{(a \cos t_n + u \cdot a \cos \vartheta_n)^2}{a^2} + \frac{(b \sin t_n + u \cdot b \sin \vartheta_n)^2}{b^2} = 1$$

Nach kurzer Rechnung ($u = 0$ liefert den Punkt $\vec{s}(t_n)$) ergibt sich:

$$u = -2(a \cos t_n \cos \vartheta_n + b \sin t_n \sin \vartheta_n) = -2 \cos(t_n - \vartheta_n)$$

Aus der Gleichung

$$\begin{pmatrix} a \cos t_{n+1} \\ b \sin t_{n+1} \end{pmatrix} = \begin{pmatrix} a \cos t_n \\ b \sin t_n \end{pmatrix} + u \cdot \begin{pmatrix} a \cos \vartheta_n \\ b \sin \vartheta_n \end{pmatrix}$$

Erhalten wir t_{n+1} :

$$t_{n+1} = \arccos(\cos t_n + u \cos \varphi_n)$$

Auf Grund der geometrischen Situation (die Kugel startet auf einem Randpunkt der Ellipse und bewegt sich in Richtung des Ellipseninneren) gibt es immer einen weiteren Schnittpunkt mit der Ellipse und somit eine Lösung der Gleichung. Insbesondere liegt $\cos t_n + u \cos \varphi_n \in [-1,1]$ und $\arccos(\cos t_n + u \cos \varphi_n)$ ist definiert.

Dabei gibt es zwei Lösungen, nämlich $t_{n+1} \in [0, \pi]$ und $2\pi - t_{n+1} \in]\pi, 2\pi[$. Welche der beiden die gesuchte ist, ergibt sich, wenn wir sie in die zweite Komponente der obigen Gleichung einsetzen.

Für den Winkel α_{n+1} betrachten wir die Ellipsentangente im Punkt $\vec{s}(t_{n+1})$:

$$\dot{\vec{s}}(t_{n+1}) = \begin{pmatrix} -a s \sin t_{n+1} \\ b \cos t_{n+1} \end{pmatrix}$$

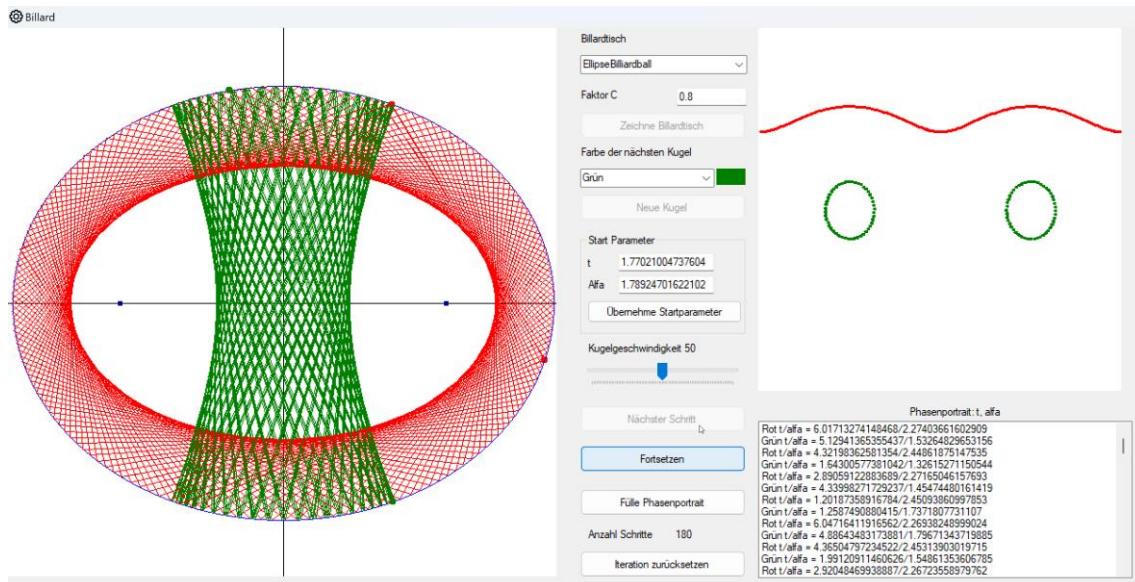
Wenn sie mit der positiven x-Achse den Winkel ψ_{n+1} einschliesst, dann gilt:

$$\alpha_{n+1} = \psi_{n+1} - \varphi_n$$

Somit hat man für das Ellipsenbillard eine Abbildung:

$$f: (t_n, \alpha_n) \rightarrow (t_{n+1}, \alpha_{n+1}), [0, 2\pi[\times]0, \pi[\rightarrow [0, 2\pi[\times]0, \pi[$$

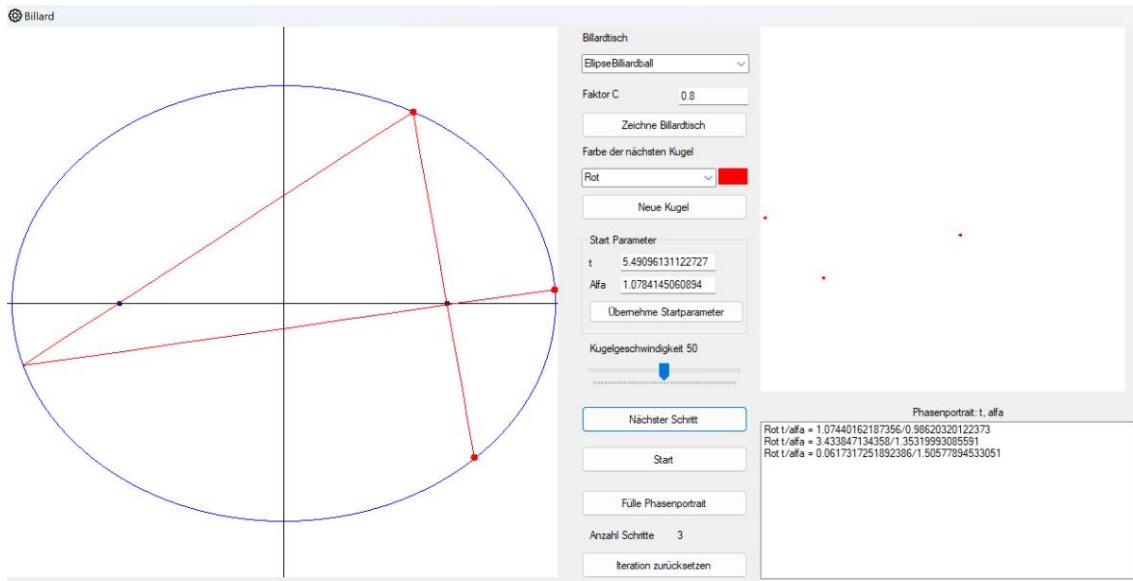
Diese Abbildung wird beim elliptischen Billard iteriert.



Links: Elliptischer Billardtisch des «Simulator». Rechts: Zugehöriges Phasenportrait

Man sieht im obigen Bild: Wenn der Startwinkel flach gewählt wird (oben rot), dann läuft die Kugelbahn um die Brennpunkte der Ellipse herum. Im Phasenportrait wird auf der horizontalen Achse der Parameter t abgetragen und vertikal der Reflexionswinkel α . Das Bild des roten Orbits ist im Phasenportrait eine Wellenlinie. Wenn der Startwinkel steil ist (oben grün), dann läuft die Kugelbahn zwischen den Brennpunkten auf und ab. Das entsprechende Bild im Phasenportrait ist ellipsenförmig.

Man weiss aus der Optik, dass Strahlen, welche von einem Brennpunkt ausgehen, im anderen Brennpunkt gebündelt werden. Ein Test mit dem «Simulator» zeigt ähnliches Verhalten:



Eine Kugel startet rechts unten und die Bahn läuft beim Start durch den Brennpunkt rechts

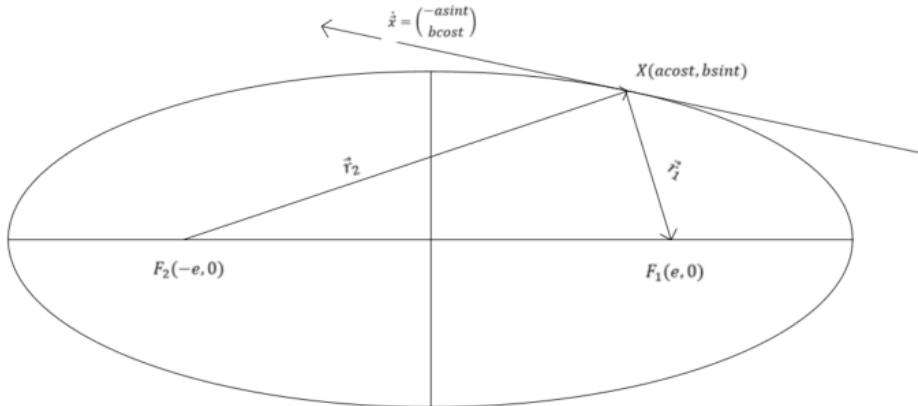
Es gibt elegante elementargeometrische Beweise für die Behauptung, dass eine Kugelbahn, welche durch den einen Brennpunkt geht, nach dem Stoss durch den anderen Brennpunkt geht. Wir wollen hier einen Beweis mit Vektorgeometrie anführen.

Die Brennpunkte einer Ellipse mit Hauptachse a und Nebenachse b haben die Koordinaten $(\pm e, 0)$ wobei $e^2 = a^2 - b^2$. Es sei t wieder der Parameter der eingangs definierten Parameterdarstellung der Ellipse.

Als Vorbereitung stellen wir fest:

$$e \cdot \cos t \leq e < \sqrt{e^2 + b^2} = a$$

Somit ist $a - e \cdot \cos t > 0$



Brennpunkte der Ellipse

Für den Betrag der Vektoren $\vec{r}_{1,2} = \begin{pmatrix} \pm acost + e \\ \pm bsint \end{pmatrix}$ gilt:

$$\begin{aligned}
 |\vec{r}_{1,2}|^2 &= a^2 \cos^2 t + b^2 \sin^2 t + e^2 \pm 2eacost \\
 &= a^2(1 + \cos^2 t) + b^2(\sin^2 t - 1) \pm 2eacost \\
 &= a^2 + a^2 \cos^2 t - b^2 \cos^2 t \pm 2eacost
 \end{aligned}$$

$$= a^2 + e^2 \pm 2eacost = (a \pm ecost)^2$$

Weil $a - e \cdot cost > 0$ gilt somit: $|\vec{r}_{1,2}| = a \pm ecost$

Nun berechnen wir:

$$\vec{r}_{1,2} \cdot \dot{\vec{x}} = -easint \mp a^2 costsint \pm b^2 sintcost = -easint \mp e^2 sintcost = -esint(a \pm ecost)$$

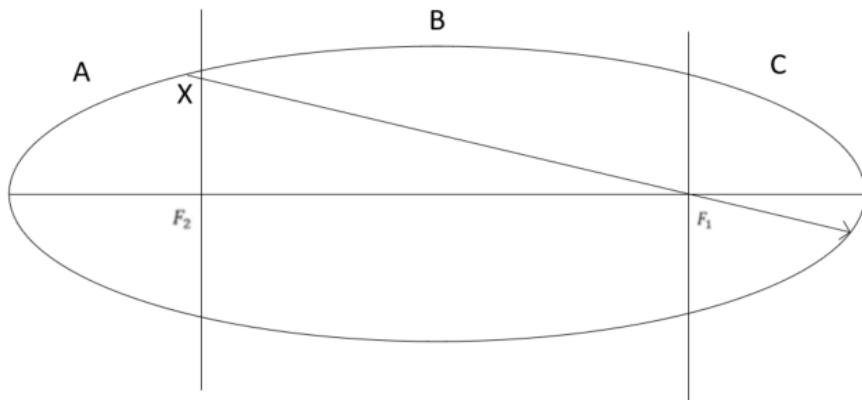
Somit gilt:

$$\cos\alpha_{1,2} = \frac{\vec{r}_{1,2} \cdot \dot{\vec{x}}}{|\vec{r}_{1,2}| |\dot{\vec{x}}|} = \frac{-esint}{|\dot{\vec{x}}|}$$

Wenn $\alpha_{1,2}$ je der Winkel zwischen $\vec{r}_{1,2}$ und der Tangente $\dot{\vec{x}}$ ist. Die rechte Seite ist aber unabhängig vom Vorzeichen bzw. unabhängig davon, ob man \vec{r}_1 oder \vec{r}_2 betrachtet und somit ist $\alpha_1 = \alpha_2$.

□

Betrachten wir nochmals den Fall einer Bahn, welche durch einen Brennpunkt geht. Dazu teilen wir den Bereich der Ellipse in drei Teile:



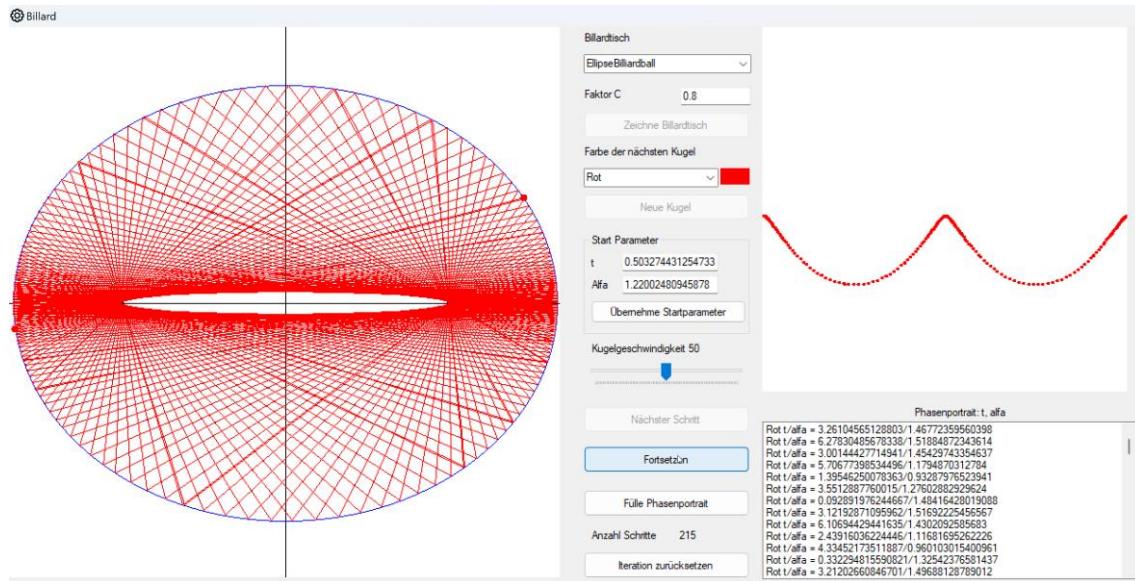
Ellipsenebene unterteilt in die senkrechten Streifen A, B, C

Fall 1: Die Kugel startet im Bereich A im Punkt X und geht dann durch den weiter entfernten Brennpunkt F_1 . Dann läuft sie nach dem Stoss durch den Brennpunkt F_2 und der nächste Stosspunkt X'' liegt näher der x-Achse als X. Bei jedem weiteren Stoss pendelt die Kugel zwischen den Bereichen A und C und jedes Mal rutscht der Stosspunkt näher an die x-Achse. Ohne dass wir den Beweis jetzt ins Detail vertiefen: Die Bahn nähert den 2-Zyklus entlang der Hauptachse der Ellipse, aber nur so weit, bis sie eine Kaustik bildet, welche sie nicht mehr schneidet.

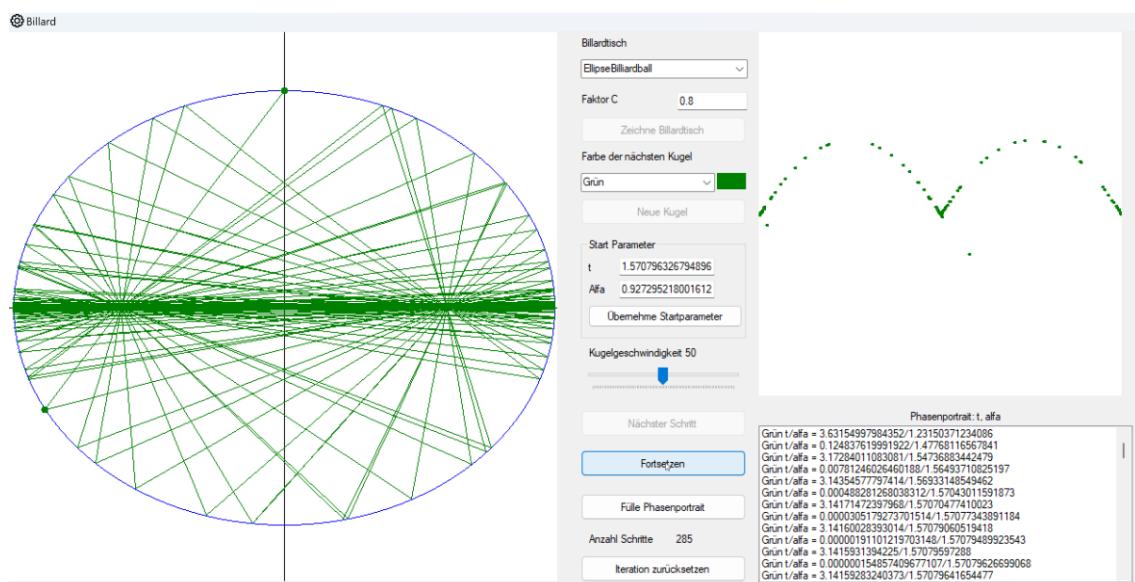
Fall 2: Die Kugel startet im Bereich B und geht o.B.d.A. durch den Brennpunkt F_2 . Dann haben wir wieder Fall 1.

Fall 3: Die Kugel startet im Bereich A und geht durch den näher gelegenen Brennpunkt F_2 . Dann liegt sie nach dem Stoss im Bereich B und wir haben Fall 2.

Fazit: Bahnen durch die Brennpunkte kommen der Hauptachse der Ellipse beliebig nahe, vorausgesetzt sie gehen wirklich präzis durch die Brennpunkte. Wenn man den Brennpunkt nicht präzis trifft, dann pendelt zwar die Bahn immer wieder in die Nähe der Hauptachse, kommt ihr aber nicht beliebig nahe, sondern entfernt sich zwischendurch wieder von ihr. Ausgespart bleibt eine Ellipse (die sogenannte Kaustik).



Die Kugel ist im ersten Quadranten gestartet und hat den linken Brennpunkt knapp verfehlt



Relativ präziser Start im Punkt $(0, b)$ mit explizit berechneten Startparametern t und α

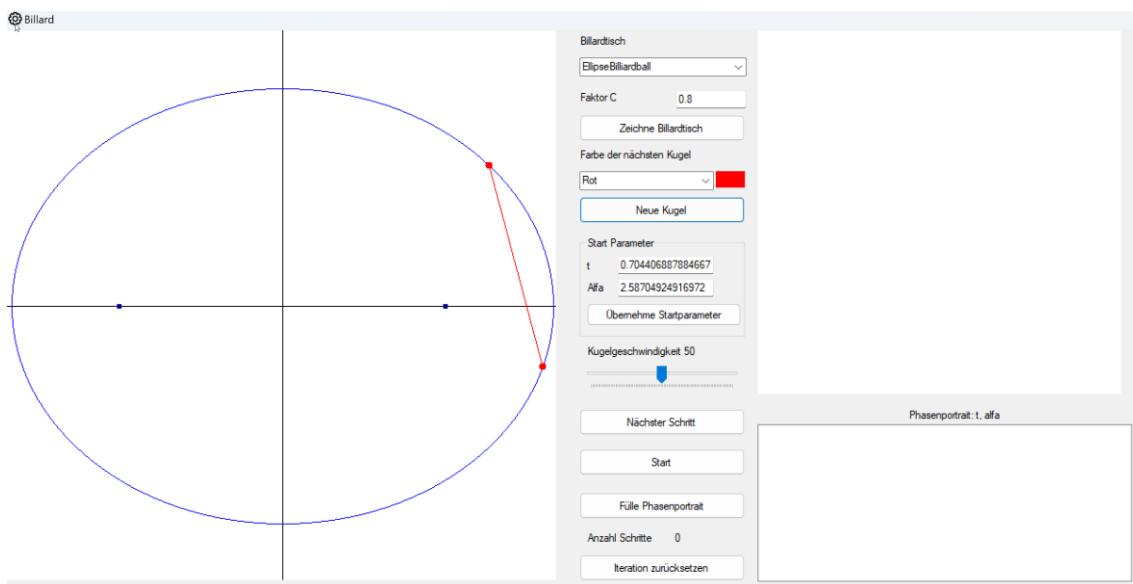
Im obigen Experiment berechnen wir möglichst präzise Startparameter. Wenn wir im Punkt $(0, b)$ starten, dann ist $t = 1.570796326794896619231321691$ (der Datentyp Dezimal unterstützt 28 Dezimalstellen). Da das Achsenverhältnis im obigen Beispiel 0.8 ist, ist $a = 1$ und $b = 0.8$, also $e = 0.6$ und der Startwinkel ist $\alpha = \arctan\left(\frac{4}{3}\right) = 0.9272952180016122324285124629$. Diese Werte geben wir manuell bei den Startparametern ein, generieren eine neue Kugel und klicken dann «Startparameter übernehmen». Das Resultat ist ein einigermaßen präzises Bild einer Kugel, welche vorerst entlang der Hauptachse pendelt, und zwar sehr nahe derselben. Mit der Zeit macht sich aber der anfängliche Fehler bei der Präzision bemerkbar und die Kugel beginnt wieder von der Hauptachse weg zu laufen, bevor sie wieder eine Zeitlang in der Nähe der Hauptachse pendelt.

4.3. Implementierung des elliptischen Billards im «Simulator»

Da immer wieder Winkel zwischen einem Vektor und der positiven x-Achse zu bestimmen sind, wird dies unterstützt durch die Funktion `CalculateAngleOfDirection (DeltaX, DeltaY)` in der Klasse `ClsMathHelperBilliard`. Übergeben werden dabei die Koordinaten des Vektors. Die Rückgabe ist ein Winkel in $[0, 2\pi[$. Verwendet wird die Funktion insbesondere zur Bestimmung der Winkel φ, ψ, ϑ sowie des Parameters t . Sie ist definiert durch die mit (1) markierte Formel im vorherigen Abschnitt.

Die Schnittstelle zwischen der `FrmBilliardtable` und der Bewegungslogik wird durch ein Interface bereitgestellt: `IBilliardball`.

`IBilliardball` verwaltet die Eigenschaften der Billardkugel (z.B. Farbe, Grösse etc.). Sie stellt Methoden zur Verfügung für die Kugelbewegung auf dem Tisch. Ebenso gibt es Methoden, welche die anfängliche Platzierung der Billardkugel sowie die Richtung des ersten Stosses ermöglicht. Diese Parameter kann der User mit der Maus positionieren.



Positionierung einer Kugel in `FrmBilliardTable`

Nachdem man die Farbe für die neue Kugel festgelegt hat (oben rot), kann man auf «Neue Kugel» klicken. Dann erscheint die Kugel standardmässig im Punkt $(0, b)$. Mit gedrückter linker Maustaste kann man nun die Startposition wählen (oben etwas rechts von $(0, b)$). Wird die Maustaste losgelassen, ist die Startposition fixiert. Wenn man die linke Maustaste zum zweiten Mal drückt (es erscheint jeweils eine Hand), kann man die Startrichtung wählen (oben nach rechts hinunter). Lässt man die Maustaste los, ist auch die Startrichtung der Kugel fixiert. Der Startparameter t und der Startwinkel α werden dann im Bereich «Start Parameter» angezeigt.

Nun kann man die nächste Kugel ebenso mit der gewünschten Farbe erzeugen und platzieren. Dank der Farben kann man die Kugeln und ihre Bahnen anschliessend unterscheiden.

Manchmal möchte man Startparameter t und Startwinkel α präzis berechnen und angeben. Dann kann man diese Werte, nachdem die Kugel erzeugt ist, im Bereich «Start Parameter» eingeben und auf «Startparameter übernehmen» klicken. Dann werden Startposition und Startwinkel der Kugel entsprechend angepasst und die Kugel positioniert.

Für das elliptische Billard gibt es die Klasse `ClsEllipseBilliardball`, welche das Interface `IBilliardball` implementiert.

Der Leser hat so die Möglichkeit durch wenig Aufwand weitere Formen des Billards zu implementieren. Er muss eine Klasse für die Billard Kugel programmieren, welche das Interface *IBilliardBall* implementiert.

Die je nach Billard verschiedene Kugelklasse (beim elliptischen Billard die Klasse *ClsEllipseBilliardball*) enthält die allgemeine Logik für die Kugelbewegung. Die Kugelbahn wird laufend in eine Bitmap gezeichnet, welche das Image der Picture Box PicDiagram ist. Die Kugel selbst wird in PicDiagram gezeichnet. Durch das refreshing der Picture Box ist jeweils nur die aktuelle Position der Kugel sichtbar, während die Bitmap inklusive Kugelbahn auf dem aktuellen Stand sichtbar ist. Um das zu unterstützen, braucht die Kugel im Konstruktor die Übergabe von PicDiagram sowie der dazugehörigen Bitmap. Da die Kugel in mathematischen Koordinaten arbeitet und erst die Klasse *ClsGraphicTool* die Umrechnung in Pixeleinheiten besorgen, brauchen letztere auch die Angabe der mathematischen Wertebereiche für x und y. Das ist das Standardquadrat [-1,1] x [-1,1]. Es wird ebenfalls der Kugel im Konstruktor übergeben.

Die Kugel unterstützt auch ihre Positionierung für den Start sowie die Anfangsrichtung für den ersten Stoss, welche mit Hilfe der Maus vorgenommen wird. Die Startparameter können auch manuell eingegeben werden.

Wenn die Iteration gestartet wird, bewegt sich die *ClsEllipseBilliardball* selbständig innerhalb der Ellipse gemäss den mathematischen Algorithmen des vorigen Abschnittes. Aus dem letzten Stosspunkt und der aktuellen Stossrichtung berechnet die Kugel zuerst den nächsten Stosspunkt. Anschliessend den Tangentenwinkel im nächsten Stosspunkt und daraus resultiert die Richtung für den nächsten Stoss.

Aus den Gleichungen

$$\varphi_{n+1} = \psi_{n+1} + \alpha_{n+1}$$

Und

$$\alpha_{n+1} = \psi_{n+1} - \varphi_n$$

Ergibt sich für die Richtung der Kugel nach dem n-ten Stoss direkt:

$$\varphi_{n+1} = 2\psi_{n+1} - \varphi_n$$

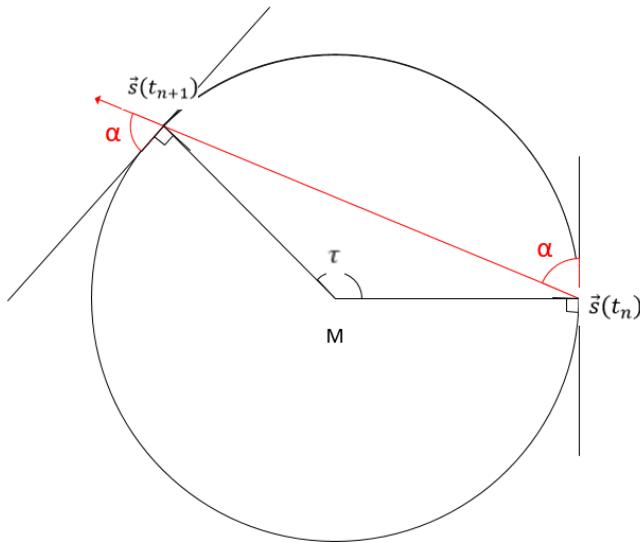
Der «Simulator» arbeitet direkt mit dieser Formel, protokolliert aber den Winkel α_n zusammen mit dem Parameter t_n im Phasenportrait.

Da die Kugel die Regeln für ihre Bewegung selbst kennt, ermöglicht dies, mehrere Kugeln zu instanziiieren und deren Bewegung parallel laufen zu lassen. Die Kugeln werden durch ihre Farben unterschieden und es sind fünf verschiedene Farben wählbar. Die Kugelgeschwindigkeit kann dabei verändert werden und wirkt sich auf alle Kugeln aus.

Gleichzeitig schreibt die Kugel die Parameter (t_n, α_n) in ein Phasendiagramm auf der rechten Fensterseite sowie in eine List Box.

4.4. Billard im Kreis

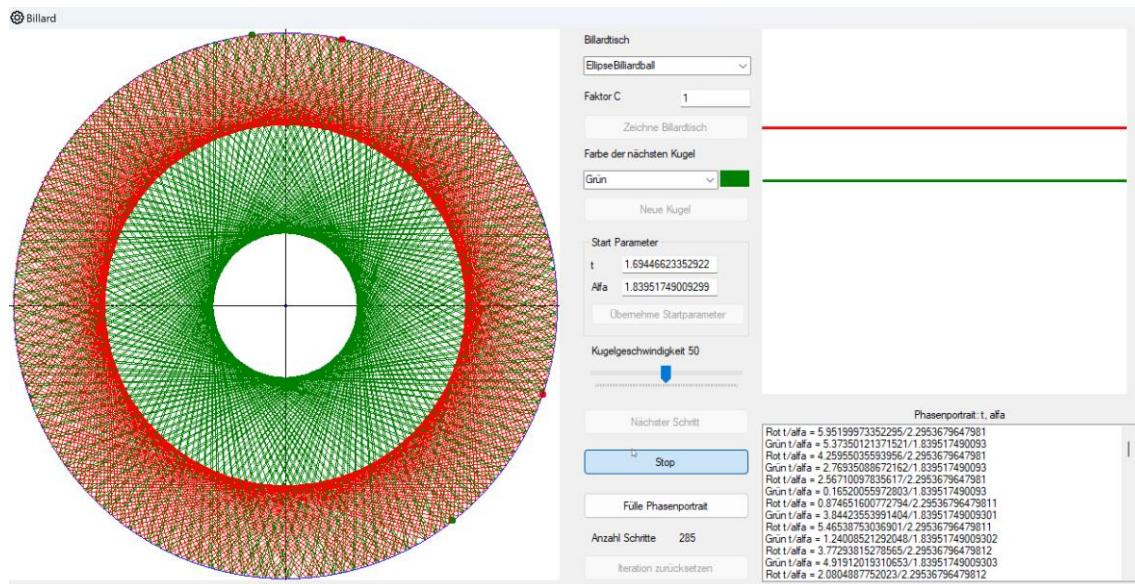
Bevor wir das elliptische Billard untersuchen, lohnt es sich, noch einen Blick auf den Spezialfall $a = b$ zu werfen, das heisst, auf das Billard im Kreis.



Billard im Kreis: Der Winkel α ist konstant. Ferner ist $\tau = 2\alpha$

Das Dreieck mit den beiden Stosspunkten und dem Mittelpunkt M ist gleichseitig, da der Abstand M von jedem Stosspunkt gleich dem Radius r ist. Somit sind die Basiswinkel dieses Dreiecks gleich. α ist je das Komplement eines Basiswinkels zu $\pi/2$, also ebenfalls gleich.

Hier ein Bild, generiert durch den «Simulator» für zwei verschiedene Kugeln: rot mit einem flachen Reflektionswinkel und grün mit einem steilen:



Billard im Kreis

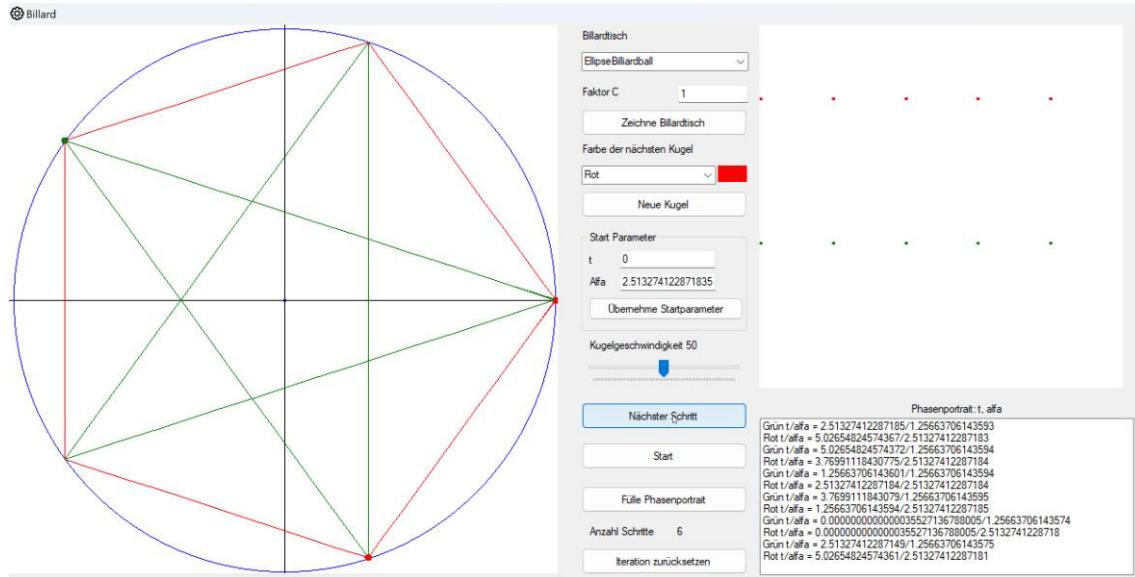
Der Winkel α ist hier konstant. Der Parameter t nimmt für den nächsten Stosspunkt um denselben konstanten Wert $\tau = 2\alpha$ zu. Somit ist das Bild der Iteration im Phasenportrait eine Gerade parallel zur t-Achse.

Beim Kreisbillard ist τ und der Parameter t für den Stosspunkt identisch. Für den Winkel φ zwischen Richtung der Kugelbahn und der positiven x-Achse gilt:

$$\varphi = \alpha + \pi/2 = \frac{\tau + \pi}{2}$$

Nun sei der Richtungsvektor φ des Bahnstarts ein rationales Vielfaches von 2π . Dann ist auch τ ein rationales Vielfaches von 2π . Nehmen wir an, es sei $\frac{p}{q}$ ein gekürzter Bruch und $\tau = 2\pi \cdot \frac{p}{q}$. Wir betrachten nur den Fall $p < q$, denn andernfalls hat der ganzzahlige Teil des Bruches nur eine Drehung um ein Vielfaches von 2π zur Folge.

Die Bahn ist periodisch, falls τ nach n Stößen ein Vielfaches von 2π ist. Das ist gerade nach q Stößen der Fall, also ist q die Periode der Bahn. Dann ist der Richtungswinkel auf den Wert $p \cdot 2\pi$ gewachsen. Falls $p < \frac{q}{2}$ ist p die Umlaufzahl der Bahn um den Mittelpunkt des Kreises im negativen Uhrzeigersinn. Falls $p > \frac{q}{2}$, dann ist $q - p$ die Umlaufzahl der Bahn um den Kreismittelpunkt im positiven Uhrzeigersinn.



Bahnen mit Periode $q = 5$ und Umlaufzahlen $p = 1$ (rot) bzw. $p = 2$ (grün)

Falls der Startwinkel φ der Bahn ein irrationales Vielfaches von 2π ist, dann ist auch der Zentriwinkel τ ein irrationales Vielfaches von 2π . Dann wird sich die Bahn nie schliessen und die Bahn ist aperiodisch. Die Punkte einer solchen Bahn liegen dicht auf dem Kreisrand. Die Begründung dafür liefert folgende Überlegung:

Sei ϑ_1 der Drehwinkel der Bahn zwischen zwei aufeinanderfolgenden Stosspunkten. Wenn wir den Einheitskreis betrachten, ist das gerade die Bogenlänge, welche zwischen zwei Stosspunkten liegt. Ausgehend von einem Startpunkt x auf dem Kreisrand führen wir dann so viele Stosse aus, bis die Bahn zu ersten Mal wieder dem Punkt x nahekommt. Das sei nach n Drehungen der Fall. Wesentlich für die Begründung ist, dass die Bahn von x den Punkt x selbst nie genau treffen wird, weil sie sonst ein rationales Vielfaches von 2π wäre.

Wir bezeichnen n Drehungen um den Winkel ϑ_1 mit $D_{\vartheta_1}^n$. Dann liegt x zwischen den Punkten:

$$D_{\vartheta_1}^{n-1}(x) < x < D_{\vartheta_1}^n(x)$$

Entweder liegt x dann näher beim Punkt $D_{\vartheta_1}^{n-1}(x)$ oder sonst näher beim Punkt $D_{\vartheta_1}^n(x)$. O.B.d.A. nehmen wir an, dass x näher bei $D_{\vartheta_1}^n(x)$ liegt. Diese Distanz ist dann kleiner als $\frac{\vartheta_1}{2}$. Somit gilt:

$D_{\vartheta_1}^n =: D_{\vartheta_2}$. Wir ersetzen also $D_{\vartheta_1}^n$ durch eine Drehung D_{ϑ_2} mit $\vartheta_2 < \frac{\vartheta_1}{2}$. Nun können wir dasselbe Argument auf ein Vielfaches der Drehung D_{ϑ_2} anwenden. Wir ersetzen also für ein gewisses m:

$$D_{\vartheta_1}^m =: D_{\vartheta_2}$$

Wenn wir dasselbe Argument auf D_{ϑ_2} anwenden, dann gibt es eine Drehung D_{ϑ_3} mit $\vartheta_3 < \frac{\vartheta_2}{2}$, welche ein Vielfaches der Drehung D_{ϑ_2} ist. Wir wiederholen dasselbe Argument genügend oft, bis wir einen beliebig kleinen Drehwinkel $\vartheta_k < \frac{\vartheta_1}{2^{k-1}}$ erhalten, wobei die Drehung um diesen Winkel ein Vielfaches von D_{ϑ_1} ist: $D_{\vartheta_k} = D_{\vartheta_1}^r$ für ein genügend grosses $r \in \mathbb{N}$. Wenn wir diese Drehung D_{ϑ_k} weiter auf x anwenden, wird die Bahn jedes beliebig kleinen Intervall treffen.

Eine verlockende Frage ist, ob man das elliptische Billard als affines Bild des Kreisbillards auffassen könnte. Das in Anlehnung an das erste Kapitel, wo wir viel Klarheit über das logistische Wachstum gewonnen haben, indem wir dies als Konjugierte der Zeltabbildung darstellen konnten. Die Zeltabbildung war leicht zu untersuchen.

Wenn also $g: (s, \beta) \mapsto (s', \beta')$ das Kreisbillard ist mit gewissen Startparametern, dann sollte das elliptische Billard $f: (t, \alpha) \mapsto (t', \alpha')$ darstellbar sein als seine Konjugierte in der Form:

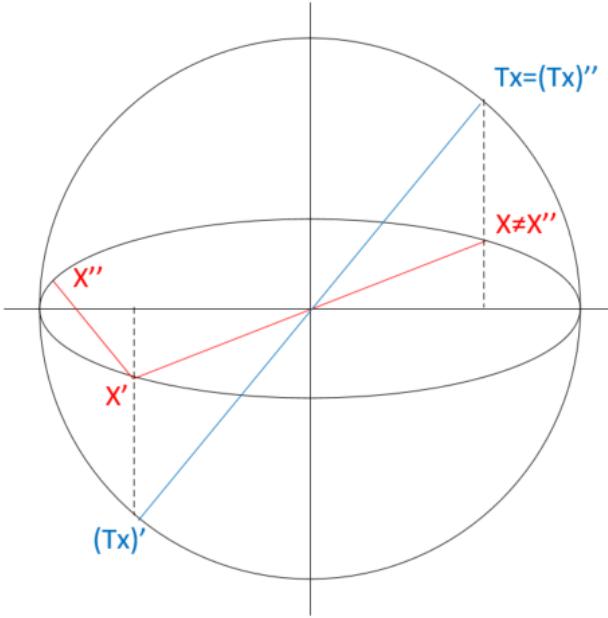
$$f(t, \alpha) = T^{-1} \circ g \circ T(t, \alpha) = T^{-1} \circ g(s, \beta) = T^{-1}(s', \beta') = (t', \alpha')$$

Wobei T die Streckung der Ebene in y-Richtung um den Faktor a/b ist. Dann nämlich wird eine Ellipse mit den Achsen a und b in einen Kreis mit Radius a überführt. Für einen Punkt $\vec{x}(t)$ auf der Ellipse gilt dann:

$$T: \vec{x}(t) = \begin{pmatrix} a \cos t \\ b \sin t \end{pmatrix} \mapsto \vec{s}(t) = \begin{pmatrix} a \cos t \\ a \sin t \end{pmatrix}$$

Wir würden also den Startpunkt und den Startwinkel der Kugel zuerst durch T auf den entsprechenden Startpunkt einer Kugel im Kreis abbilden. Dann wird im Kreis ein Stoss durchgeführt, was zu einem nächsten Stoßpunkt führt, wobei beim Kreis der Startwinkel beim Stoss konstant bleibt. Anschliessend transformieren wir alles zurück auf die Ellipse und hoffen dann, dass wir dasselbe Resultat erhalten, wie wenn wir den Stoss gemäss Reflexionsgesetz direkt in der Ellipse durchgeführt hätten.

Kann das gutgehen? Zum Beispiel bleiben die Brennpunkte der Ellipse bei der Abbildung T fix, verlieren aber im Kreis ihre Bedeutung. Wir suchen also eher ein Gegenbeispiel, welches zeigt, dass die vorgeschlagene Konjugation so nicht funktioniert. Dazu betrachten wir folgende Figur:



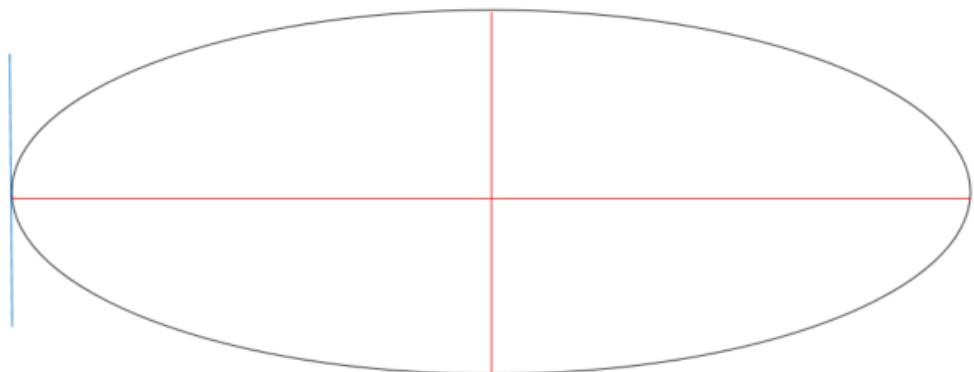
Affine Abbildung zwischen Ellipse und Kreis

Wir betrachten den Punkt X. Der wird beim Billard nach zwei Stößen bei X'' landen. Das Bild von X unter der affinen Abbildung T ist TX . Dieser Punkt landet nach zwei Stößen wieder bei ihm selbst und nicht bei TX'' .

4.5. Periodische Punkte beim Ellipsenbillard

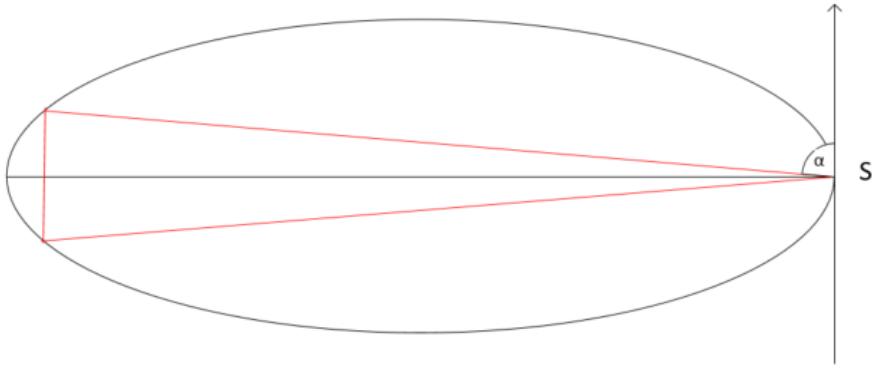
Jeder Punkt auf dem Ellipsenrand, dessen Startrichtung gerade die Richtung der Ellipsentangente ist, kann als (entarteter) Fixpunkt angesehen werden.

Bei einem 2-periodischen Punkt muss die Richtung der einfallenden Kugelbahn gerade um den Winkel π gedreht werden, damit der ausfallende Strahl wieder den ursprünglichen Ausgangspunkt trifft. Das heisst, dass die Einfallsrichtung senkrecht auf der Tangente im Stoßpunkt ist. Dasselbe gilt auch, wenn die Kugel wieder am Ausgangspunkt auftrifft. Die einzigen möglichen Bahnen sind somit:



2-periodische Bahnen beim Ellipsenbillard

Bahnen der Periode drei existieren. Wenn man nachfolgende Skizze betrachtet:



Eine drei-periodische Bahn

Ausgehend vom Punkt S ist α so zu wählen, dass die Kugel nach dem ersten Stoss senkrecht nach unten läuft und nach dem Zweiten Stoss wieder im Punkt S eintrifft. Wenn man mit dem «Simulator» experimentiert, sieht man, dass dies möglich ist. Wird α etwas zu klein gewählt, landet man etwas unterhalb von S. Wird α zu gross gewählt, landet man oberhalb von S. Wird α dazwischen stetig verändert, dann muss es ein α geben, dass der Punkt S wieder getroffen wird.

Analytisch oder elementargeometrisch scheint es schwierig, eine Lösung für das Problem zu finden. Ein vektorgeometrischer Ansatz führt nämlich auf eine komplizierte Gleichung für das gesuchte α .

Aber wir können versuchen, mit dem «Simulator» wenigstens eine Näherungslösung zu finden. Dazu starten wir mit einem etwas zu kleinen α_1 und wählen dann das nächste α_2 etwas zu gross. Das gesuchte α muss also aus Stetigkeitsgründen irgendwo dazwischenliegen. Wir benützen nun folgenden Algorithmus:

Nehmen wir an, wir haben zwei Winkel α_{n-1} und α_n , wobei α_{n-1} zu klein ist (d.h. man gelangt nach zwei Stössen nicht ganz zum Punkt S) und α_n zu gross ist. Dann finden wir das nächste Winkelpaar folgendermassen:

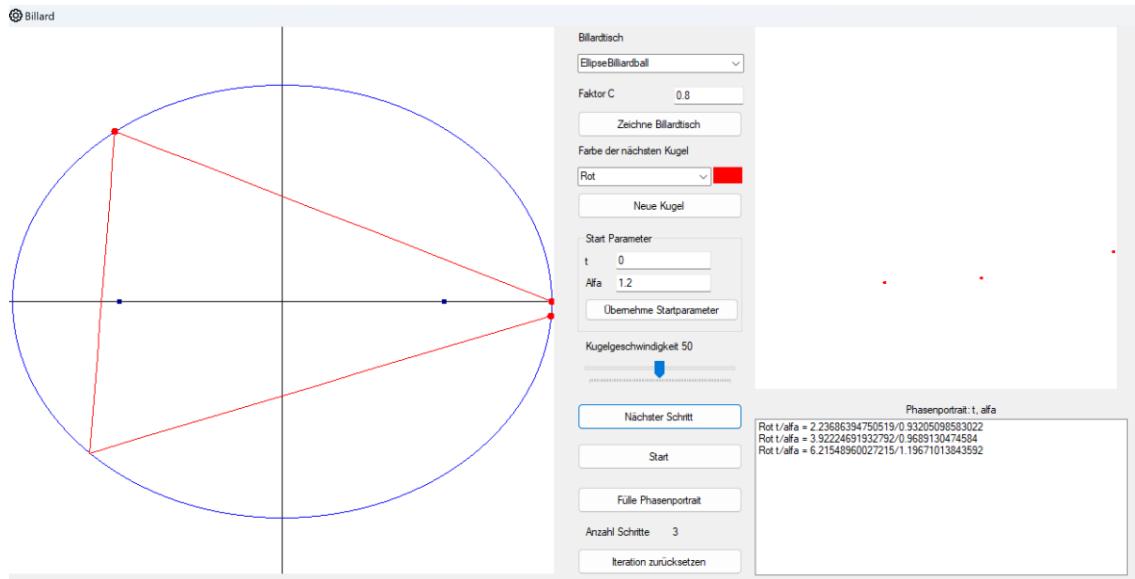
$$\delta := \frac{\alpha_{n-1} + \alpha_n}{2}$$

Dann setzen wir:

$$\begin{cases} \alpha_{n+1} = \delta, \alpha_n = \alpha_{n-1}, & \text{falls } \delta \text{ zu gross ist} \\ \alpha_{n+1} = \alpha_n, \alpha_n = \delta, & \text{falls } \delta \text{ zu klein ist} \end{cases}$$

Wieder ist der gesuchte Winkel $\alpha \in [\alpha_n, \alpha_{n+1}]$ aber diese Intervalllänge ist beim Schritt halbiert worden. Wenn wir genügend Lange iterieren, kommen wir dem gesuchten Wert immer näher.

Nachfolgend eine Tabelle mit einigen auf diese Art experimentell gefundenen Werten für den Faktor $c = 0.8$:



$\alpha_1 = 1.2$ ist etwas zu klein

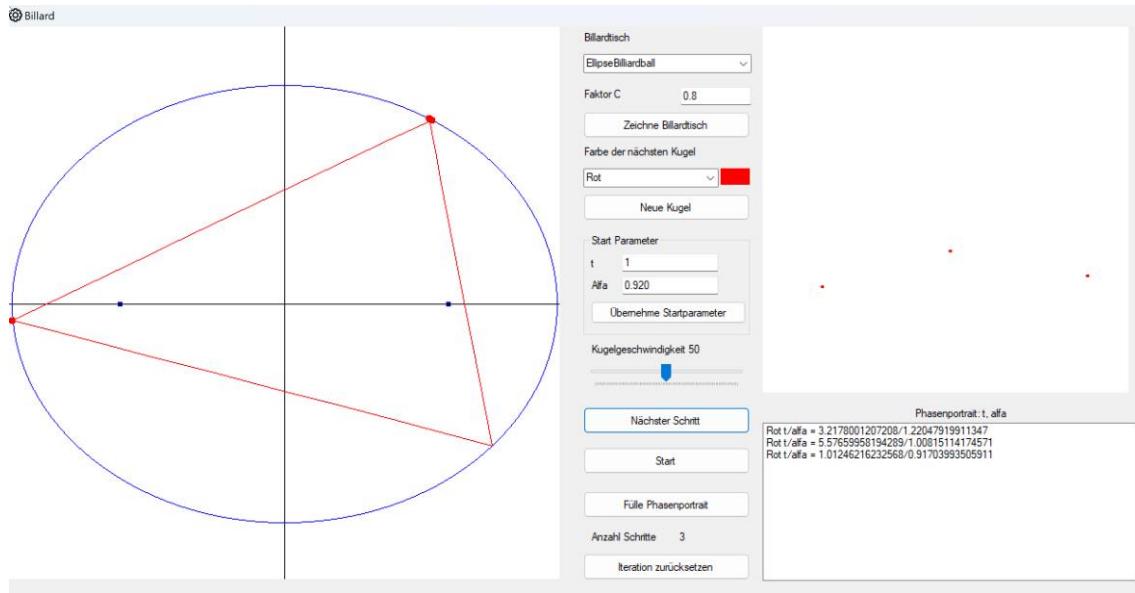
Dass $\alpha_1 = 1.2$ etwas zu klein ist, sieht man hier noch optisch. Die bessere Kontrolle hat man aber rechts im Protokoll des Parameters t . Dieser ist knapp unter 2π . Wie man ebenso feststellt, ist $\alpha_2 = 1.25$ etwas zu gross. Wir halbieren dann dieses Intervall und sehen, dass der Wert 1.225 noch immer zu gross ist. Also ersetzen wir 1.25 durch 1.225. Auf diese Art erhalten wir sukzessive eine Intervallschachtelung:

α_n	α_{n+1}
1.2	1.25
1.2	1.225
1.2125	1.225
1.21875	1.225
1.221875	1.225
1.221875	1.2234375
1.22265625	1.2234375

Erste Schritte der Intervallschachtelung, um den gesuchten Startwert α zu finden

Diese konvergiert sehr langsam. Aber immerhin ist plausibel, dass ein Grenzwert und somit ein 3-Zyklus existiert.

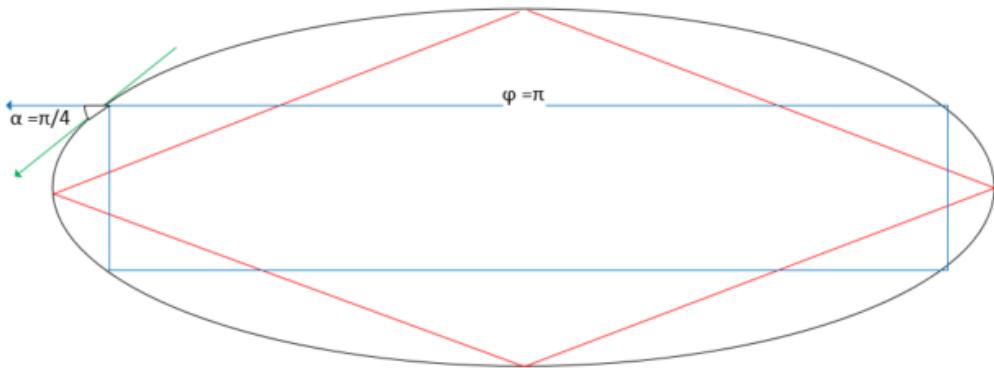
Mit derselben Methode kann man weitere, nicht symmetrische Startpositionen finden, welche zu einer drei-periodischen Bahn führen. Zum Beispiel für den Parameter $t = 1$ und $\alpha = 0.920$ als erste Näherung:



Startparameter $t = 1, \alpha = 0.920$

Weitere Versuche zeigen: Das gesuchte α für eine drei-periodische Bahn mit dem Startpunkt für den Parameter $t=1$ existiert und liegt im Intervall $[0.919, 0.920]$.

Bahnen der Periode 4 findet man leicht:



Bahnen der Periode vier

Rot dargestellt ist eine Bahn, welche von einem Scheitelpunkt zum nächsten läuft. Sie kann im Uhrzeigersinn oder Gegenuhrzeigersinn durchlaufen werden.

Blau dargestellt ist eine weitere 4-periodische Bahn. Der Aufschlagpunkt beim Winkel α ist leicht zu bestimmen: Gemäss Abschnitt 4.1 muss der neue Richtungswinkel φ nach dem Stoss senkrecht nach unten zeigen. Der Richtungswinkel der Kugel vor dem Stoss ist gleich π . Dann muss die Richtung der Tangente relativ zur positiven x-Achse $5\pi/4$ betragen. Das liefert für den gesuchten Stosspunkt die Koordinaten

$$\vec{r} \approx \begin{pmatrix} -a \cdot 0.707 \\ b \cdot 3.927 \end{pmatrix}$$

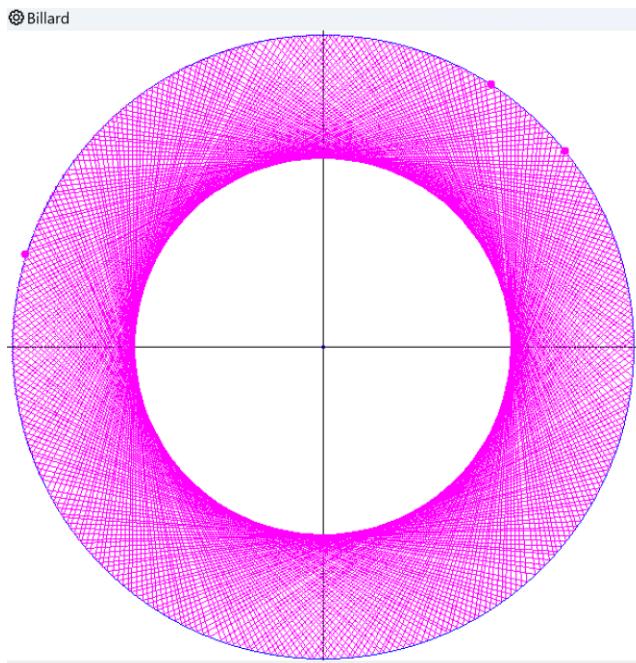
Experimente mit dem «Simulator» zeigen, dass offenbar auch nicht symmetrische Bahnen der Periode 4 existieren, z.B. mit Startpunkt $t=1$ liegt ein entsprechendes α im Intervall $[0.721, 0.722]$.

Mit dieser Methode lassen sich offenbar Startwinkel finden, welche zu Bahnen einer beliebig vorgegebenen Periode führen.

Wir werden später einen entsprechenden Satz für beliebige streng konvexe Billardtische kennen lernen.

4.6. Kaustik im elliptischen Billard

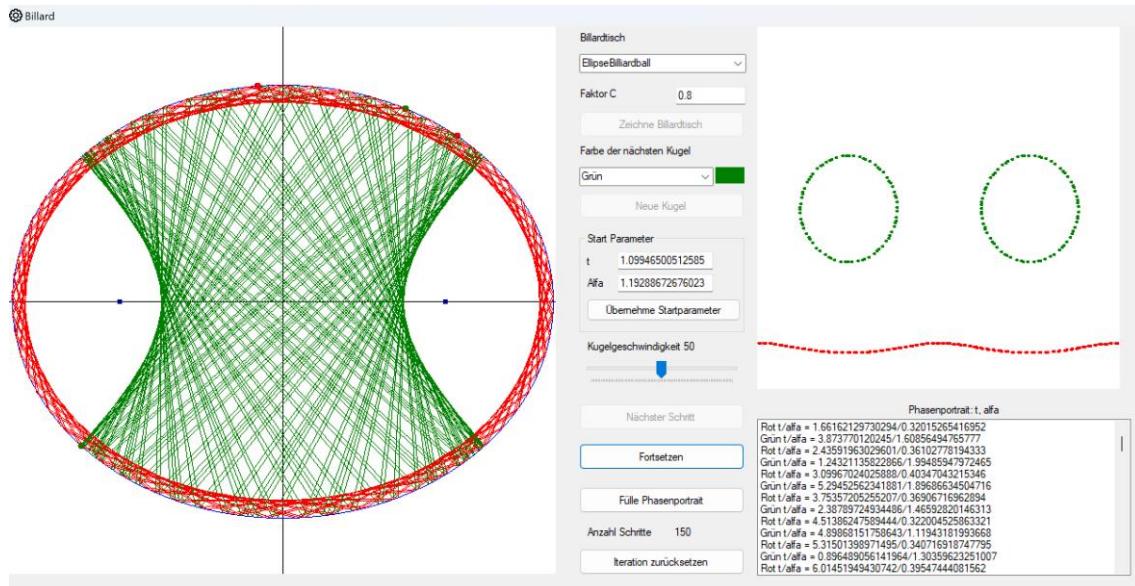
Eine *Kaustik* ist ein Begriff aus der Optik und bezeichnet einen Bogen, an welchen Lichtstrahlen eines optischen Systems Tangenten sind. Im Falle des Billards betrachtet man statt Lichtstrahlen die Kugelbahnen. Beim Kreis sind diese tangential zu einem konzentrischen Kreis:



Die Kaustik beim Billard im Kreis ist ein konzentrischer Kreis

Die einzelnen Abschnitte der Kugelbahn sind Kreissehnen. Da der Winkel α zwischen Kugelbahn und Tangente im Stoßpunkt für alle Bahnabschnitte konstant ist, sind alle zur Bahn gehörenden Kreissehnen gleich lang. Deren Berührungs punkt mit der Kaustik ist der Sehnenmittelpunkt. Wenn wir eine solche Kreissehne drehen, beschreibt deren Mittelpunkt einen konzentrischen Kreis mit Radius $\cos\alpha$.

Beim elliptischen Billard ist es ähnlich:

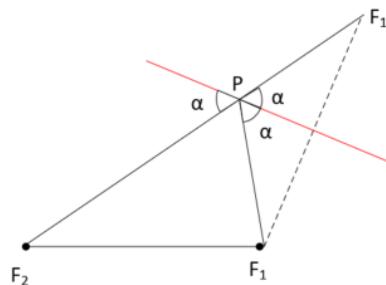


Kaustiken im elliptischen Billard

Gezeigt werden im obigen Bild die Bahnen von zwei unterschiedlichen Kugeln, eine in grün mit Startrichtung zwischen den Brennpunkten, und eine in rot mit Startrichtung ausserhalb der Brennpunkte der Ellipse. Man sieht, dass die Kaustik in beiden Fällen offenbar wieder ein Kegelschnitt ist. Im roten Fall eine Ellipse und im grünen Fall eine Hyperbel. Diese haben dieselben Brennpunkte wie die Ellipse des Billardtisches. Rechts ist das jeweilige Phasendiagramm in der entsprechenden Farbe dargestellt.

Wir wollen diesen Sachverhalt im Falle der roten Bahnkurve beweisen. Mit Hilfe der Vektorgeometrie kann das aufwendig werden. Wir greifen auf einen elementargeometrischen Beweis zurück.

Zuerst müssen wir uns überlegen, wo ein Bahnabschnitt die Kaustik berührt. Wir richten das so ein, dass eine Kugel, welche in einem Brennpunkt startet, und am Bahnabschnitt gespiegelt wird, zum anderen Brennpunkt läuft. Dabei muss das Reflexionsgesetz gelten.



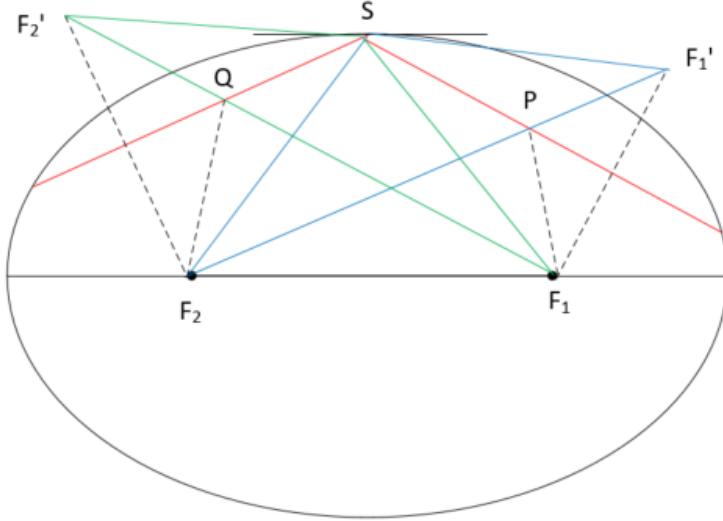
Konstruktion des Punktes P auf dem Bahnabschnitt (rot)

Oben ist der Bahnabschnitt rot dargestellt. Wir müssen den gesuchten Treffpunkt P so wählen, dass für einen Brennstrahl das Reflexionsgesetz gilt. Dazu spiegeln wir F_1 am Bahnabschnitt und verbinden dann den gespiegelten Punkt F_1' mit dem anderen Brennpunkt F_2 . Im Schnittpunkt P mit dem Bahnabschnitt sind alle markierten Winkel gleich und somit gilt das Brechungsgesetz, falls die Kaustik den Bahnabschnitt im Punkt P berührt. Es gibt also eine Ellipse mit Brennpunkten F_1 und F_2 , so dass der Bahnabschnitt eine Tangente an diese Ellipse ist.

Man kann leicht prüfen, dass wir denselben Punkt P erhalten, wenn wir statt F_1 den anderen Brennpunkt F_2 gespiegelt hätten.

Nun müssen wir zeigen, dass für den nachfolgenden Bahnabschnitt der nächste Treffpunkt Q , der analog konstruiert wird, auf derselben Ellipse ist wie P .

Dazu betrachten wir folgende Figur:



Zwei aufeinanderfolgende Bahnabschnitte (rot) mit dem Stosspunkt S und ihre Berührungsstellen P und Q mit der gesuchten Ellipse, welche ebenfalls die Brennpunkte F_1 und F_2 hat

Zuerst stellen wir fest, dass das grün markierte Dreieck S, F_1, F_2' und das blau markierte Dreieck S, F_1', F_2 kongruent sind. Es gilt nämlich: $|SF_2'| = |SF_2|$ und $|SF_1'| = |SF_1|$. Ferner ist der Winkel an der Spitze S für jedes Dreieck derselbe wegen dem Reflexionsgesetz einerseits und weil der Winkel $\angle(F_1SF_2)$ für beide Dreiecke derselbe ist. Somit ist auch die dritte, dem Punkt S gegenüberliegende Seite, bei beiden Dreiecken gleich lang und es gilt:

$$|F_2'F_1| = |F_1'F_2| \Rightarrow |F_2Q| + |QF_1| = |F_2P| + |PF_1|$$

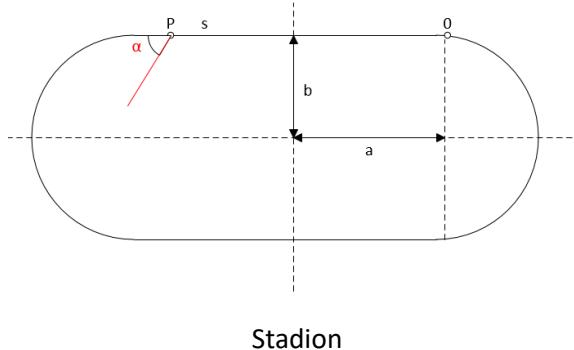
Die Ellipse, welche die Kugelbahn in Q berührt ist also identisch mit der Ellipse, welche die Kugelbahn in P berührt.

Eine analoge Betrachtung für den Fall, dass die Bahnabschnitte zwischen den Brennpunkten verlaufen, werden wir in Form einer Übung führen und sehen, dass in diesem Fall als Kaustik eine Hyperbel mit den Brennpunkten F_1 und F_2 erscheint.

Im konkreten Fall kann diese Ellipse (bzw. Hyperbel) berechnen, indem man wie in der obigen Konstruktion vorgeht und die jeweilige Hauptachsenlänge $|F_2P| + |PF_1|$ berechnet.

Nachdem der Startpunkt und der erste Reflexionswinkel einer Kugel auf dem elliptischen Billardtisch festgelegt ist, ist auch die Kaustik definiert. Das heisst, dass der weitere Orbit nur noch vom jeweils nächsten Stosspunkt abhängig ist. Man legt von einem Stosspunkt einfach eine Tangente an die Kaustik, schneidet sie mit der Ellipse und erhält den nächsten Stosspunkt. Dabei hat man zwei Möglichkeiten für die Tangente, wenn der Orbit zwischen den Brennpunkten verläuft. Die Bewegung hängt dann nur vom Parameter t ab, wenn der Orbit ausserhalb der Brennpunkte verläuft. Diese Information ergibt sich aber bereits aus dem Plot im Phasenraum.

4.7. Billard im Stadion



Das Stadion ist zusammengesetzt aus einem Rechteck mit Breite $2a$ und Höhe $2b$. An den vertikalen Seiten wird je ein Halbkreis mit Radius b angefügt. Als Koordinatensystem verwenden wir eines mit Nullpunkt im Mittelpunkt des Stadions und den Symmetriegeraden als Achsen.

Eine Billardkugel wird am Rand des Stadions gemäss Reflexionsgesetz reflektiert.

Für die Parametrisierung fixieren wir einen Nullpunkt auf dem Rand, und zwar den Punkt (a, b) . Um einen Punkt P auf dem Rand zu beschreiben, verwenden wir die Bogenlänge s zwischen diesem Punkt und dem festgelegten Nullpunkt auf dem Rand. Dabei rechnen wir modulo dem Umfang des Stadions, also modulo $L = 4a + 2\pi b$. Damit ist jeder Stoßpunkt der Billardkugel eindeutig definiert durch den Parameter s . Als zweiten Parameter verwenden wir wieder den Winkel α zwischen Kugelbahn und Kurventangente im Stoßpunkt. Wenn die Kugel auf einer Rechteckseite stösst, ist diese Tangente gerade identisch mit der Rechteckseite. Im anderen Fall hat man als Kurventangente die Kreistangente im Stoßpunkt.

Die Kugelbewegung wird also durch eine Abbildung beschrieben:

$$f: [0, 4a + 2\pi b] \times [0, \pi] \rightarrow [0, 4a + 2\pi b] \times [0, \pi]: (s, \alpha) \mapsto (s', \alpha')$$

Im Phasenportrait werden diese beiden Parameter bei jedem Stoß protokolliert.

Für die Implementierung werden wir zu jedem s die Koordinaten (x, y) des entsprechenden Randpunktes P brauchen. Das geschieht mit Hilfe einer Fallunterscheidung (2):

1. $0 \leq s \leq 2a \Rightarrow P(x, y) = (a - s, b)$
2. $2a < s < 2a + b\pi \Rightarrow P(x, y) = (-a - b \cdot \sin \frac{s-2a}{b}, b \cdot \cos \frac{s-2a}{b})$
3. $2a + b\pi \leq s \leq 4a + b\pi \Rightarrow P(x, y) = (s - 3a - b\pi, -b)$
4. $4a + b\pi < s < 4a + 2b\pi \Rightarrow P(x, y) = (a + b \cdot \sin \frac{s-4a-b\pi}{b}, -b \cdot \cos \frac{s-4a-b\pi}{b})$

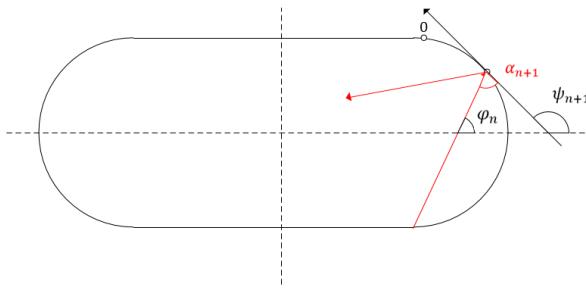
Dabei ist s modulo L gerechnet.

Für die Berechnung des jeweiligen nächsten Stoßpunktes und Stoßwinkels brauchen wir weitere Parameter. Wir verwenden dieselben Bezeichnungen wie beim elliptischen Billard:

ψ_n = der Winkel zwischen (gerichteter) Kurventangente im n-ten Stoßpunkt und positiver x-Achse

α_n = der Reflexionswinkel beim n-ten Stoß

φ_n = der Winkel zwischen dem (gerichteten) n-ten Bahnabschnitt und der positiven x-Achse



Winkel beim (n+1) -ten Stoss

Dann gilt: $α_{n+1} = ψ_{n+1} - φ_n$ und $φ_{n+1} = ψ_{n+1} + α_{n+1}$, wie beim elliptischen Billard. Daraus folgt unmittelbar:

$$φ_{n+1} = 2ψ_{n+1} - φ_n$$

Wenn wir den Stosspunkt für den (n+1) -ten Stoss kennen, können wir die Tangente im Stosspunkt berechnen und deren Winkel $ψ_{n+1}$ mit der positiven x-Achse bestimmen. Daraus ergibt sich unmittelbar der Winkel $φ_{n+1}$ den der nächste Bahnabschnitt mit der positiven x-Achse einschliesst. Für das Protokoll berechnen wir: $α_{n+1} = ψ_{n+1} - φ_n$.

Um den Tangentenwinkel $ψ$ zu bestimmen, braucht es wieder eine Fallunterscheidung (3):

1. $0 \leq s \leq 2a \Rightarrow ψ = π$
2. $2a < s < 2a + bπ \Rightarrow ψ = \frac{s-2a}{b}$
3. $2a + bπ \leq s \leq 4a + bπ \Rightarrow ψ = 0$
4. $4a + bπ < s < 4a + 2bπ \Rightarrow ψ = \frac{s-4a-bπ}{b}$

Am Start ist der erste Reflexionswinkel $α_1$ gegeben. Ebenfalls der erste Stosspunkt, so dass wir die zugehörige Tangente und ihren Winkel $ψ_1$ berechnen können. Der Winkel des ersten Bahnabschnittes ist dann $φ_1 = ψ_1 + α_1$. Mit dem zweiten Stosspunkt berechnen wir dessen Tangentenwinkel $ψ_2$ und dann $φ_2 = 2ψ_2 - φ_1$. Damit erhalten wir während der Iteration alle weiteren Winkel.

Es bleibt noch ein Verfahren zum Berechnen des jeweils nächsten Stosspunktes zu finden. Ausgehend von einem Stosspunkt P mit Ortsvektor $→p = \begin{pmatrix} u \\ v \end{pmatrix}$ ist der nächste Bahnabschnitt der Kugel auf einer Geraden mit Winkel $φ$ zur positiven x-Achse, also auf einer Geraden:

$$→x(t) = \begin{pmatrix} u \\ v \end{pmatrix} + t \cdot \begin{pmatrix} \cos φ \\ \sin φ \end{pmatrix}, t \in \mathbb{R}$$

Falls $φ \neq 0, π$ schneiden wir diese Gerade zuerst mit den Geraden $y = ±b$. Das liefert

$$t = \frac{\pm b - v}{\sin φ}$$

Wenn $u + t \cos φ \in [-a, a]$ dann haben wir den Schnittpunkt gefunden.

Falls wir den Schnittpunkt noch nicht gefunden haben, suchen wir ihn auf den beiden Halbkreisen mit Mittelpunkt $(±a, 0)$ und Radius b .

Diese haben die Gleichungen:

$$(x \mp a)^2 + y^2 = b^2$$

Mit der Bedingung $x > a$ für den rechten Kreis (mit negativem Vorzeichen für a in der Klammer) und $x < -a$ für den linken Kreis mit positivem Vorzeichen in der Klammer.

Für den gesuchten Punkt auf der Geraden liefert das die Bedingung:

$$(u + t \cos \varphi \mp a)^2 + (v + t \sin \varphi)^2 = b^2$$

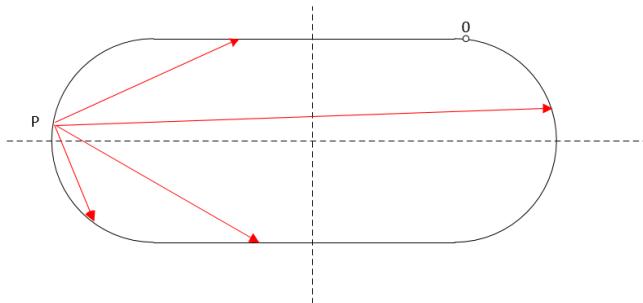
Nach einer kurzen Rechnung erhalten wir für t :

$$t = -B \pm \sqrt{B^2 - C}$$

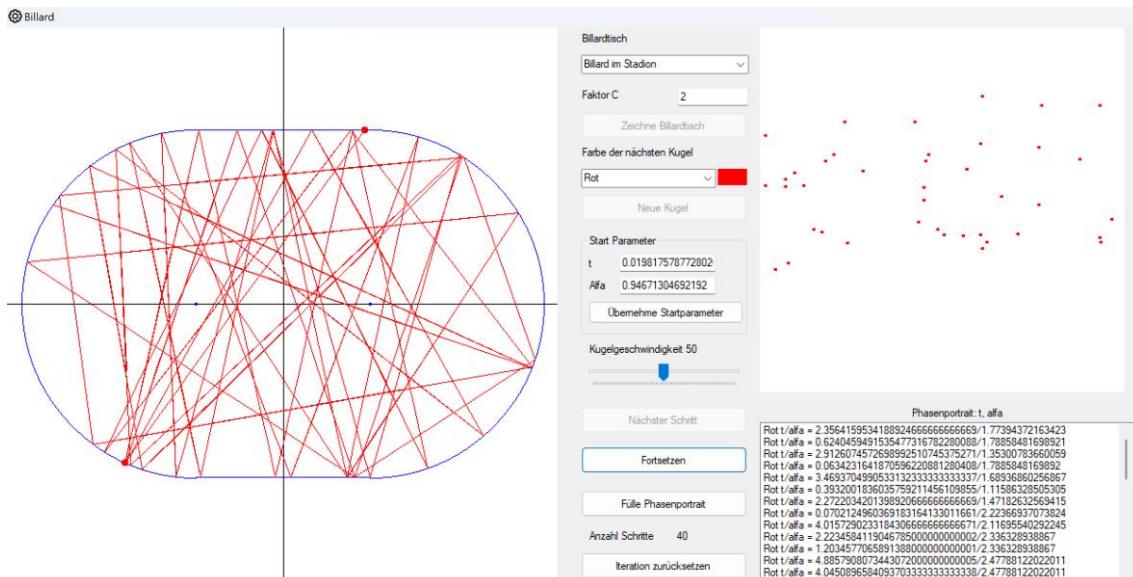
Wobei

$$B = (u \mp a) \cos \varphi + v \sin \varphi$$

$$C = (u \mp a)^2 + v^2 - b^2$$

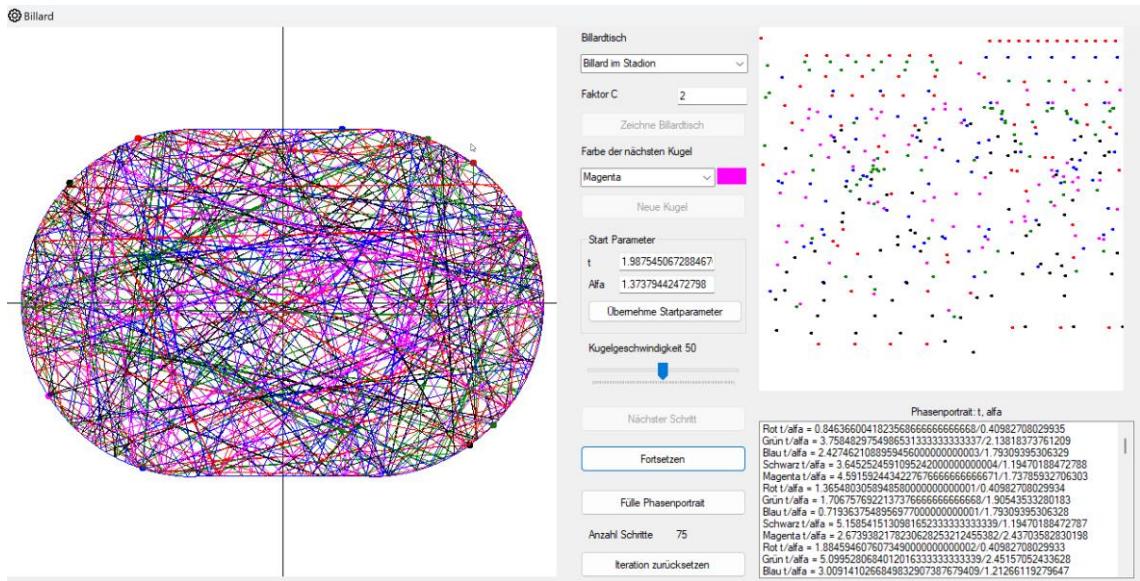


Verschiedene Szenarien zur Berechnung des nächsten Stosspunktes



Die ersten 40 Stöße einer Kugel im Stadion

Nachfolgend wurden fünf Kugeln verschiedener Farbe gleichzeitig gestartet und haben je 75 Stöße durchgeführt.



Das Billard im Stadion ist chaotisch

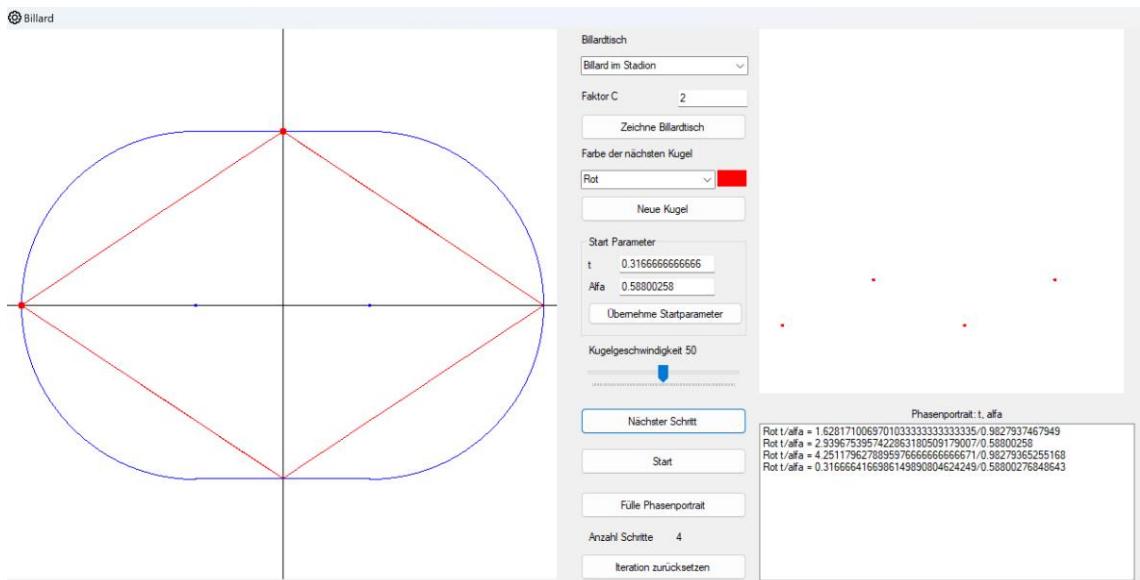
Wie man rechts im Phasenportrait sieht, sind die Einträge über das ganze Phasenportrait verteilt. Beim elliptischen Billard waren die Einträge im Phasenportrait auf gewisse Kurven beschränkt und mindestens die Transitivität ist beim elliptischen Billard als Abbildung

$$f: (t_n, \alpha_n) \rightarrow (t_{n+1}, \alpha_{n+1}), [0, 2\pi[\times]0, \pi[\rightarrow [0, 2\pi[\times]0, \pi[$$

nicht erfüllt, da gewisse Bereiche im Phasenportrait immer ausgespart bleiben.

Hingegen wurde für das Billard im Stadion chaotisches Verhalten nachgewiesen. Diese Art von Billard ist auch bekannt als «Bunimowich Stadium» nach Leonid Bunimowich, welcher diesen Nachweis erbracht hat.

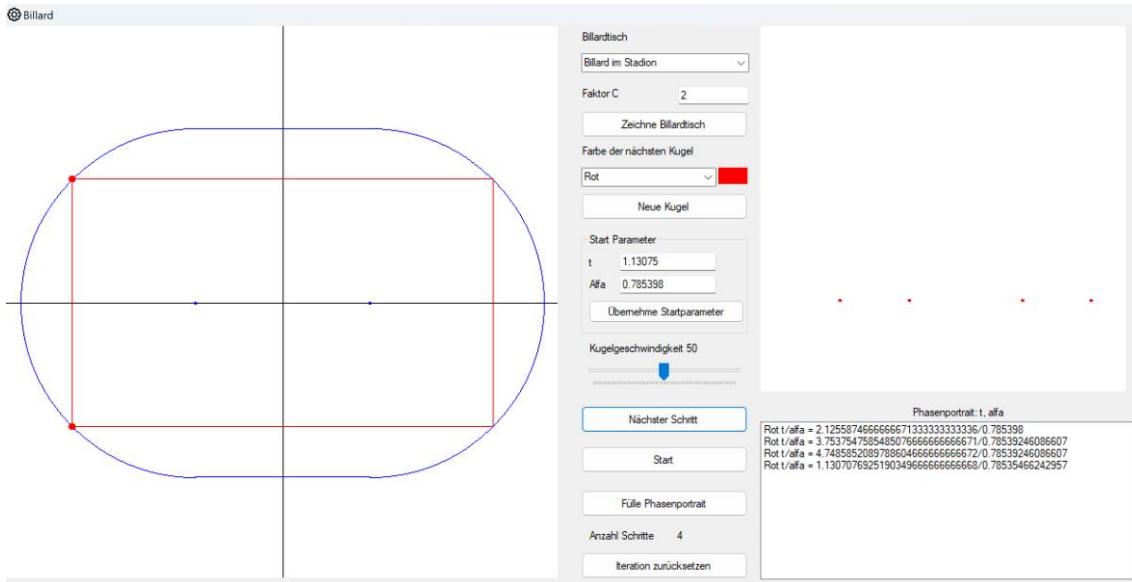
Auch für das Billard im Stadion ist es einfach, periodische Punkte zu finden. Zum Beispiel gibt es ausgehend von einem Scheitelpunkt $(0, a + b)$ Bahnen jeder Periode $2k$, $k \in \mathbb{N}$, welche durch entsprechende Bahnen im Kreis gefunden werden können.



Eine Bahn der Periode 4

Für den Faktor $c = 2$ ist $a = 0.31\bar{6}$, $b = 0.6\bar{3}$. Der Parameter für den Startpunkt ist $s = 0.31\bar{6}$. Für den Startwinkel α gilt $\alpha = \arctan \frac{b}{a+b} = 0.58800258 \dots$

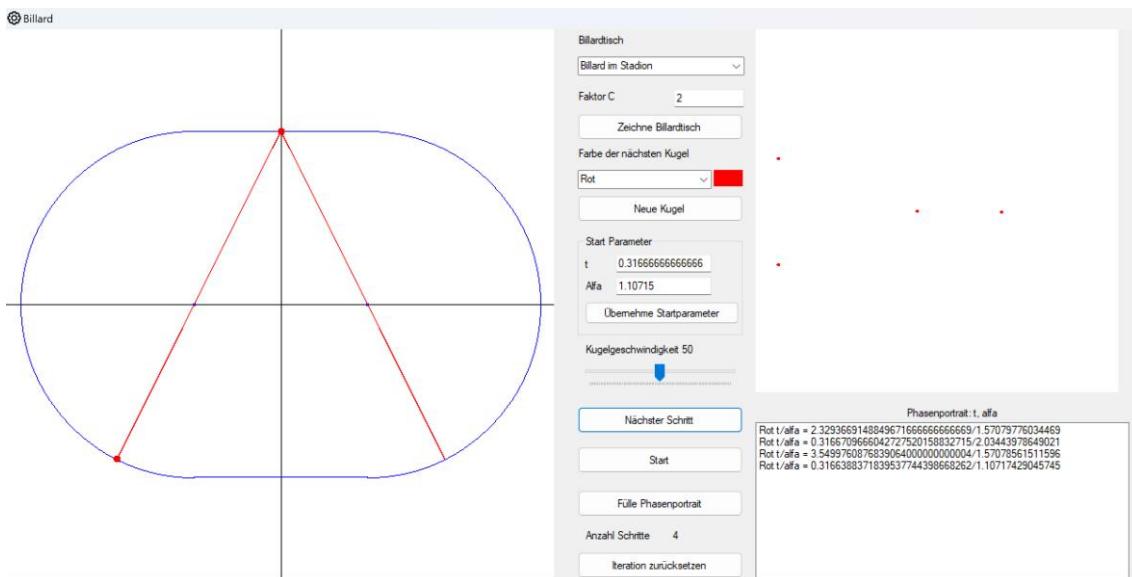
Eine weitere Bahn mit Periode vier ist die folgende:



Eine alternative Bahn der Periode vier

Wie man leicht nachrechnen kann, gilt für obige Bahn $s = 2a + b \cdot \frac{\pi}{4}$ und $\alpha = \frac{\pi}{4}$.

Wenn eine Bahn durch den Mittelpunkt eines Kreises geht und anschliessend an diesem gespiegelt wird, läuft sie auf dem Rückweg wieder durch den Mittelpunkt. Damit kann man weitere periodische Bahnen finden.

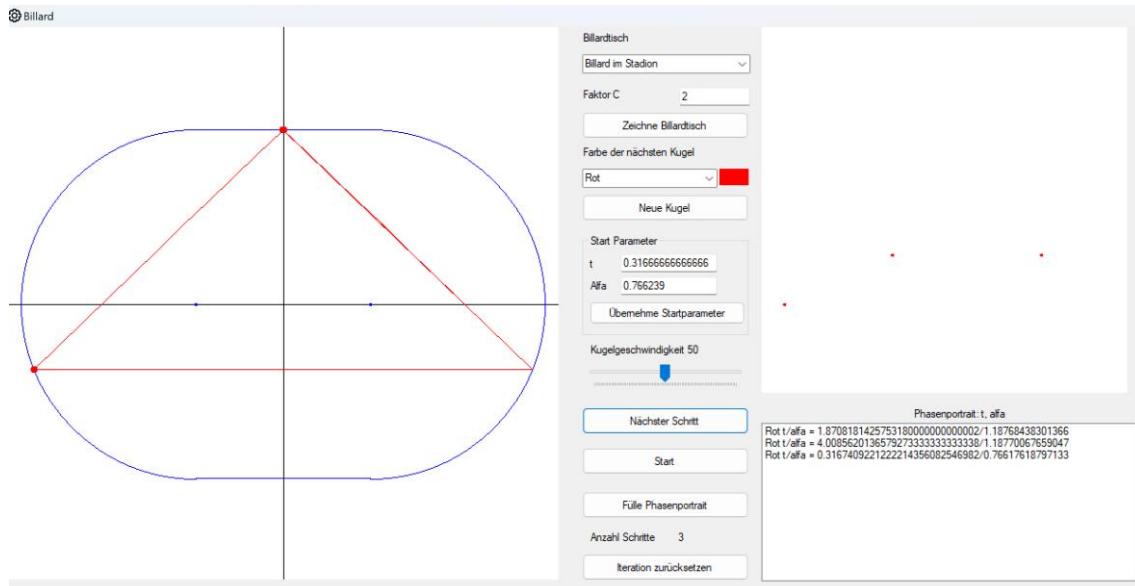


Eine weitere Bahn der Periode vier

Im obigen Fall ist wieder $s = 0.31\bar{6}$, d.h. der Startpunkt ist $(0, b)$. Für den Winkel α gilt $\alpha = \arctan \frac{b}{a} = 1.10715 \dots$

Für Bahnen ungerader Periode können wir wieder experimentell durch dasselbe Näherungsverfahren wie schon beim elliptischen Billard die entsprechenden Startparameter wenigstens näherungsweise bestimmen. Als Beispiel suchen wir eine dreiperiodische Bahn mit Startpunkt in $(0, b)$ für den Faktor $c = 2$. Für diesen Faktor ist der Startparameter $s = a = 0.31\bar{6}$. Wir starten zuerst mit einem Winkel $\alpha = 0.75$ und passen diesen laufend so an, dass nach drei Stößen im Protokoll wieder der Parameter $s = 0.31\bar{6}$ für die Position der Kugel möglichst gut erreicht wird.

Nach einigen Iterationsschritten erhalten wir als Näherung $\alpha \approx 0.766239$:



Angenäherte Startparameter für eine dreiperiodische Bahn

4.8. Implementierung im «Simulator»

Die Klasse `ClsStadiumBilliardball` implementiert das Interface `IBilliardball`. Im Fenster `FrmBilliardtable` muss nichts angepasst werden.

Die Billardkugel enthält die gesamte Bewegungslogik, nur so können wir auf einfache Art beliebig viele Kugeln instanziiieren. Damit die Bezeichnung des Parameters für den Stoßpunkt einheitlich ist, verwenden wir in der Implementierung für diesen Parameter wieder die Bezeichnung t statt s . Die Fallunterscheidung (2) für die Berechnung der Koordinaten eines Randpunktes aus dem Parameter t wird in der Klasse `ClsStadiumBilliardball` implementiert, und zwar unter der Bezeichnung `CalculateMathPointFromT`.

Wie beim elliptischen Billard kann der Startpunkt und der Startwinkel der Kugel am Anfang mit der Maus gesetzt werden.

Die Fallunterscheidung (3) zur Berechnung des Tangentenwinkels ψ ist in `CalculatePsi` implementiert.

4.9. Ovales Billard

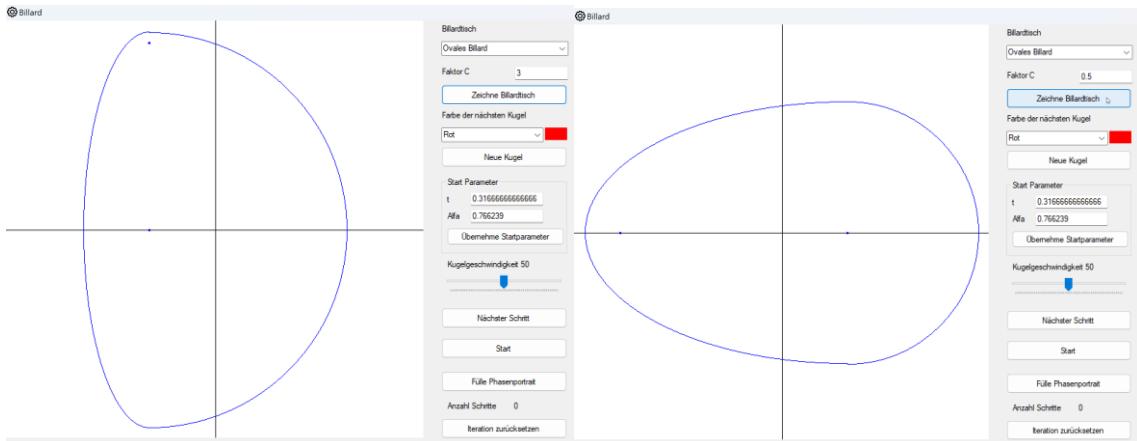
Einige Hühner hätten wahrscheinlich keine Freude an gewissen Eiern, welche wir im Folgenden als Billardtisch benutzen. Wir setzen den Billardtisch nämlich zusammen aus einem Halbkreis und einer halben Ellipse. Dabei ist b immer der Kreisradius und falls $a \geq b$ ist a die Hauptachse der Ellipse und

b deren Nebenachse. Falls $a < b$ ist es gerade umgekehrt. Diese Parameter werden nicht explizit gesetzt, sondern lediglich der Faktor $c := a/b$. Dann werden a und b so bestimmt, dass die Figur gut ins Diagramm passt. Der horizontale Durchmesser des Ovals ist $a + b$, der vertikale $2b$. Dieses Maximum wird wegen der besseren Sichtbarkeit auf 1.9 gesetzt. Der Wertebereich der mathematischen Koordinaten des Diagramms ist $[-1,1] \times [-1,1]$, hat also den Durchmesser 2.

Wenn $a \geq b$ ist, dann gilt $a(1 + c) = a + b \geq 2b$, man setzt also $a = \frac{1.9}{1+c}$. Wenn $b > a$ ist, dann gilt $2ac = 2b \geq a + b$ und man setzt $a = \frac{1}{2c}$. In beiden Fällen ist dann $b = ca$.

Das wird im «Simulator» bei der Übergabe des Faktors c implementiert.

Um das Diagramm maximal auszunützen, wird dann als Mittelpunkt von Kreis bzw. Ellipse der Punkt mit der x-Koordinate $(a - b)/2$ gesetzt.



Links ein «Ei» bzw. ein Oval mit Faktor $c = 3$, rechts mit Faktor $c = 0.5$

Falls $c = 1$ ist, hat man einen Kreis.

Als Parameterdarstellung des Ovals wählen wir:

$$\vec{s}(t) = \begin{cases} \left(\begin{matrix} m + b \cos t \\ b \sin t \end{matrix} \right), & t \in [-\pi/2, \pi/2] \\ \left(\begin{matrix} m + a \cos t \\ b \sin t \end{matrix} \right), & t \in]\pi/2, 3\pi/2[\end{cases}$$

Wobei $m = (a - b)/2$ die x-Koordinate des Mittelpunktes ist. (Seine y-Koordinate ist 0).

Die so parametrisierte Kurve ist stetig. Es gilt:

$$\lim_{t \rightarrow \pi/2^+} \vec{s}(t) = \left(\begin{matrix} m \\ b \end{matrix} \right) = \vec{s}(\pi/2) \quad \text{bzw.} \quad \lim_{t \rightarrow 3\pi/2^-} \vec{s}(t) = \left(\begin{matrix} m \\ -b \end{matrix} \right) = \vec{s}(3\pi/2)$$

Sie ist aber an den Übergangsstellen nicht differenzierbar, falls $a \neq b$:

$$\lim_{t \rightarrow \pi/2^+} \dot{\vec{s}}(t) = \left(\begin{matrix} -a \\ 0 \end{matrix} \right) \neq \left(\begin{matrix} -b \\ 0 \end{matrix} \right) = \dot{\vec{s}}(\pi/2) \quad \text{bzw.} \quad \lim_{t \rightarrow 3\pi/2^-} \dot{\vec{s}}(t) = \left(\begin{matrix} a \\ 0 \end{matrix} \right) \neq \left(\begin{matrix} b \\ 0 \end{matrix} \right) = \dot{\vec{s}}(3\pi/2)$$

Wir brauchen aber jeweils nur die Steigung der Tangente in jedem Punkt und diese ist an den Übergangsstellen 0 wie auch der beidseitige limes an diesen Stellen. Die *Steigung* der Tangente ist somit stetig in diesen Punkten.

Alles Weitere ist identisch mit dem elliptischen Billard, ausser dass man bei der Parametrisierung an Stelle der Ellipse die Fallunterscheidung Ellipse/Kreis hat, je nach Parameter t . Bei der

Implementierung lagern wir die Bestimmung der Koordinaten eines Randpunktes in Abhängigkeit vom Parameter t in eine entsprechende Funktion `CalculateMathPointFromT` aus.

Alle Beziehungen zwischen den Winkeln sind dieselben:

$$\begin{cases} \alpha_{n+1} = \psi_{n+1} - \varphi_n \\ \varphi_{n+1} = \psi_{n+1} + \alpha_{n+1} \\ \varphi_{n+1} = 2\psi_{n+1} - \varphi_n \end{cases}$$

Bei der Berechnung des neuen Stosspunktes starten wir bei einem Stosspunkt $\vec{s}(t_n)$. Der Richtungswinkel des nächsten Bahnabschnittes ist φ_n . Dann liegt der gesuchte nächste Stosspunkt auf der Geraden

$$\vec{x}(t) = \begin{pmatrix} p \\ q \end{pmatrix} + u \cdot \begin{pmatrix} \cos \varphi_n \\ \sin \varphi_n \end{pmatrix}, u \in \mathbb{R}$$

mit $\begin{pmatrix} p \\ q \end{pmatrix} = \vec{s}(t_n)$.

Wenn wir diese Gerade mit dem Oval schneiden, erhalten wir den gesuchten nächsten Stosspunkt.

Im Falle der Ellipse lautet die entsprechende Gleichung:

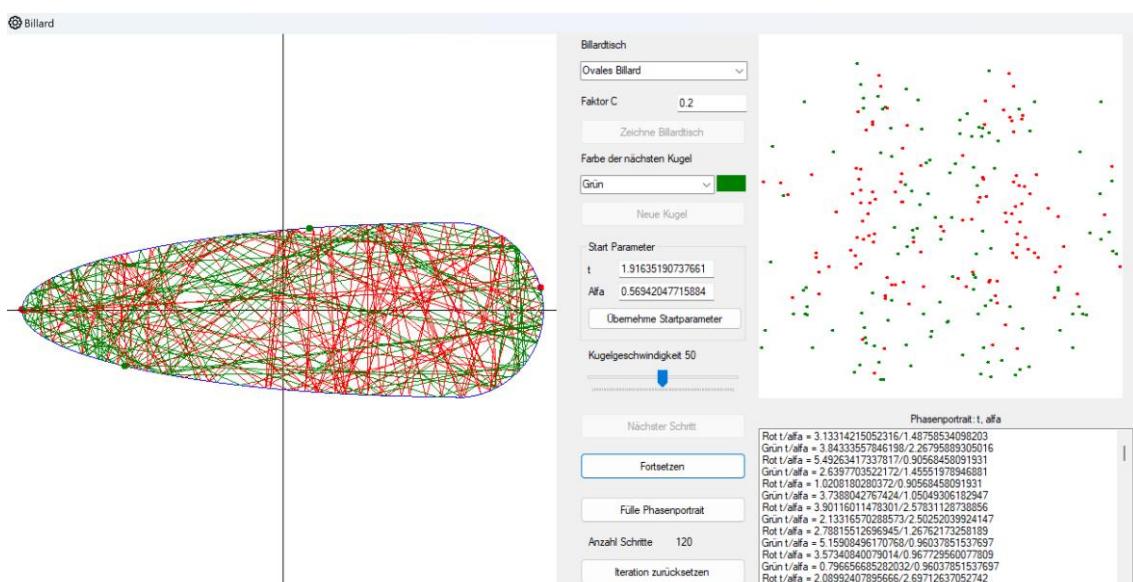
$$\frac{(x - m)^2}{a^2} + \frac{y^2}{b^2} = 1$$

Wobei $m = (a - b)/2$.

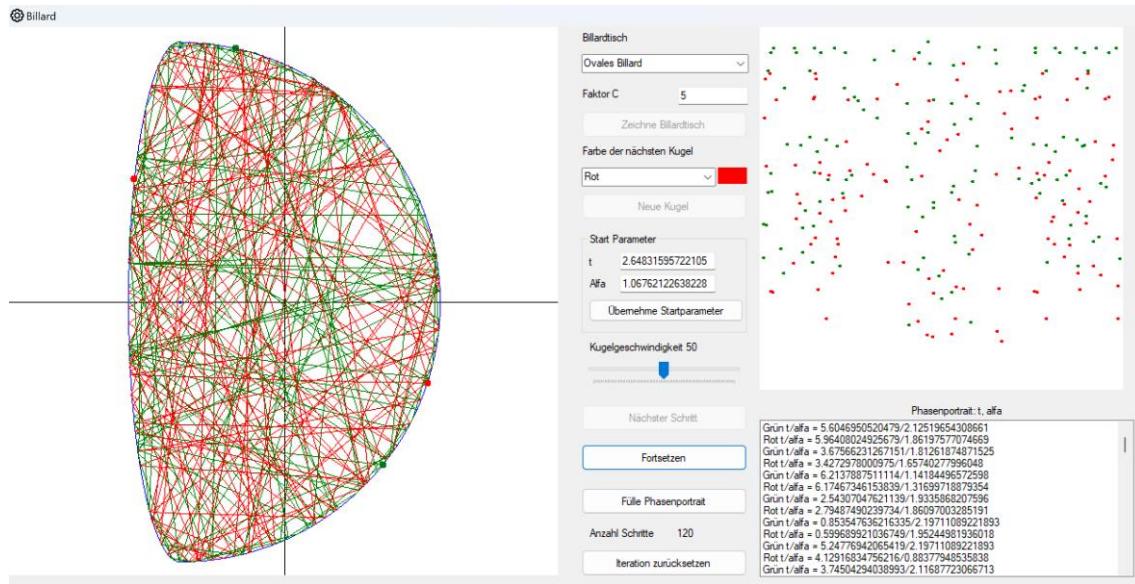
Die Lösung der nun auftretenden quadratischen Gleichung überlassen wir hier dem Leser. Im Falle des Kreises ist in obiger Gleichung $a = b$ und man geht analog vor.

Das liefert im Allgemeinen 4 Schnittpunkte, je zwei mit der Ellipse und zwei mit dem Kreis. Eine davon ist der Ausgangspunkt, also die Lösung $u = 0$, die man ausschliessen kann.

Für die Kreisschnittpunkte muss gelten, dass deren x-Koordinate $\geq m$ ist. Sonst können sie ausgeschlossen werden. Analog muss für die Ellipsenschnittpunkte gelten, dass deren x-Koordinate $< m$ ist, sonst können sie ausgeschlossen werden. Da die Gerade ausgehend von einem Randpunkt des Ovals nicht tangential ist, erhält man so genau einen zweiten Schnittpunkt.



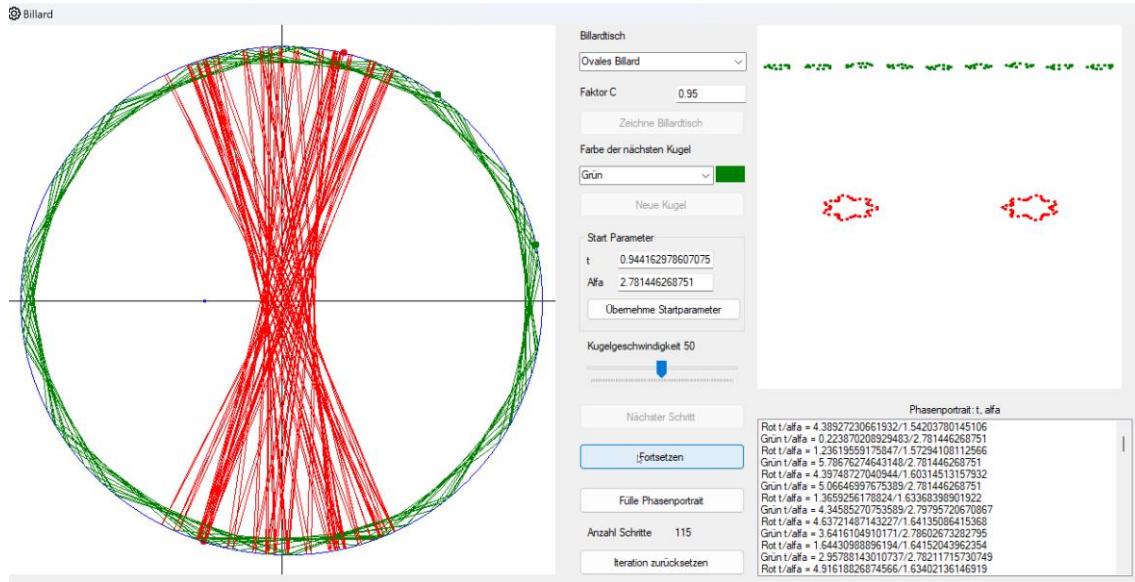
Die Bahn zweier Kugeln mit $c = 0.2$



Zwei Kugeln mit $c = 5$

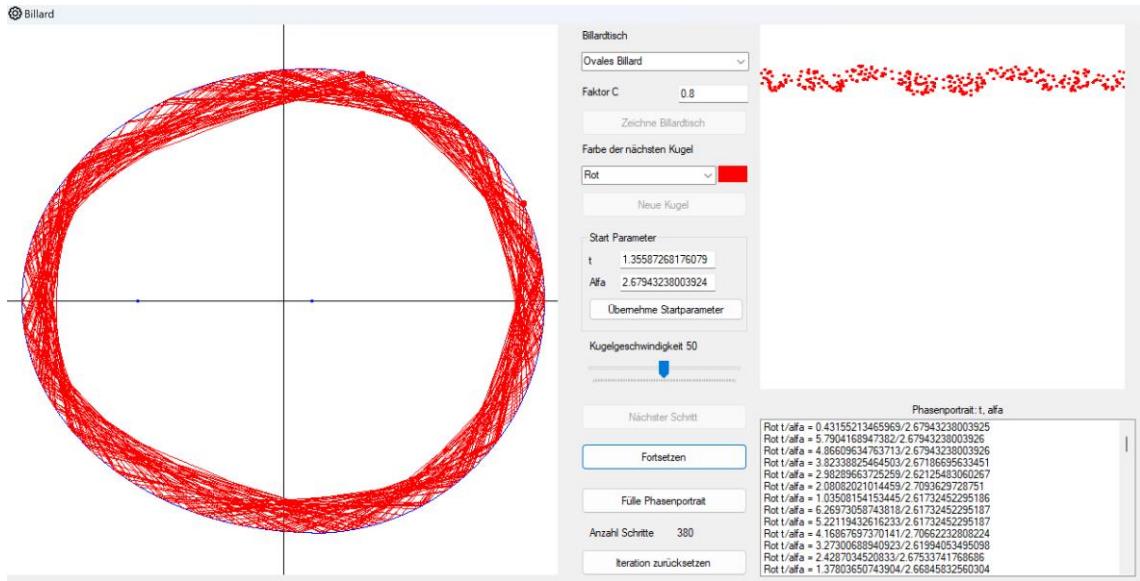
Wenn man die Kugeln länger laufen lässt, dann verteilen sich die Punkte im Phasenportrait immer mehr über den ganzen Raum. Das könnte auf chaotisches Verhalten schliessen lassen.

Wenn man aber c in der Nähe von 1 wählt, dann ähneln die Bahnen eher denen in einem elliptischen Billardtisch:



Zwei Kugeln mit $c = 0.95$

Wenn man die Kugelbahn flach startet, dann scheint die Kugel sich zuerst recht ordentlich zu verhalten, fast wie bei einer Ellipse. Nach einiger Zeit «kippt» die Kugel aber immer mehr in Richtung chaotischem Aussehen.

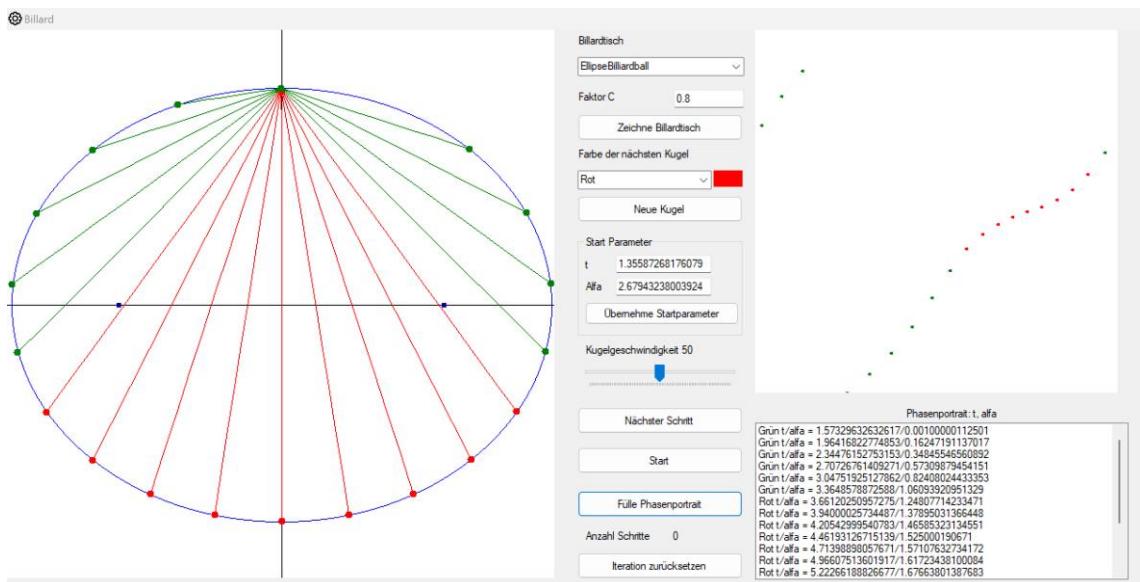


Eine Kugelbahn mit $c = 0.8$, welche nach 380 Stößen noch immer einer Kaustik nahekommt

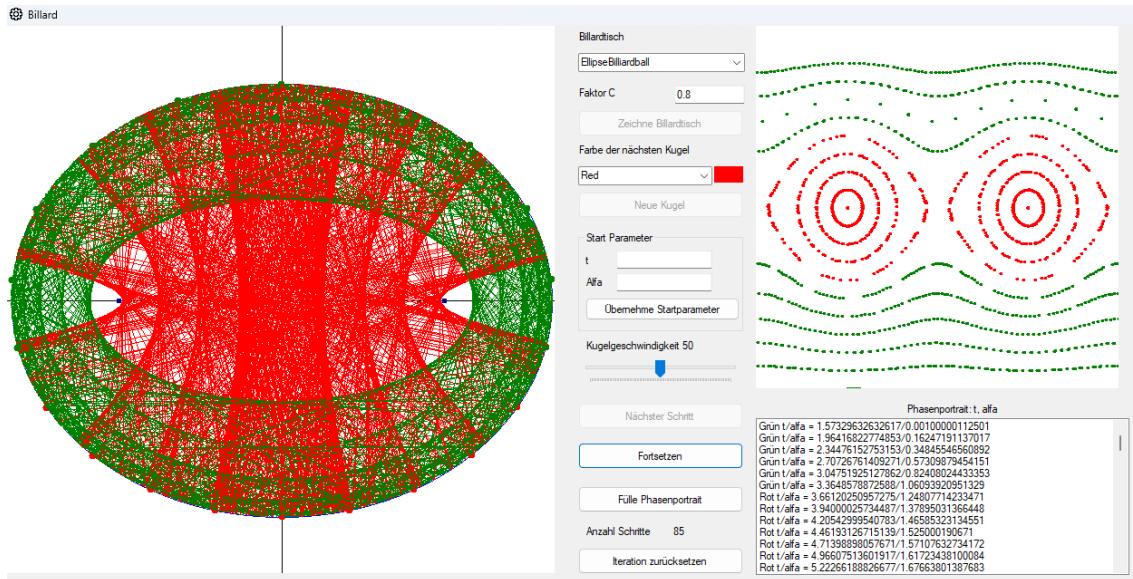
Wie bei früheren Billardtischen kann man auch beim ovalen Billard die periodischen Punkte diskutieren. Wir überlassen das dem Leser in Form einer Übung.

4.10. Darstellung des Phasenraumes

Die Schaltfläche «Fülle Phasenraum» ermöglicht es, ausgehend von vielen Kugeln mit verschiedenen Startbedingungen deren Bahnen im Phasenraum gleichzeitig darzustellen. Das gibt einen Überblick über die Komplexität des Systems.



Startposition der automatisch generierten Kugeln



Phasenraum rechts, nachdem jeder Ball 85 Stöße durchgeführt hat

4.11. Der konvexe Billardtisch

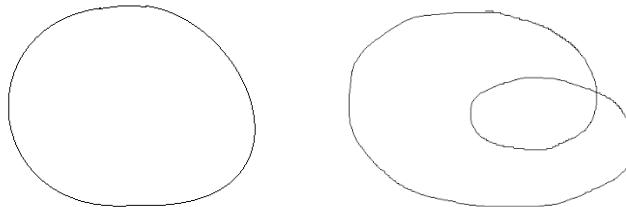
Als Ausblick betrachten wir einen Billardtisch, dessen Rand eine (ebene) konvexe, geschlossene und glatte Kurve ist. Hier lassen sich einige elementarmathematische Aussagen machen, vor allem als Anwendung des Satzes vom Maximum und Minimum von stetigen Funktionen auf einem abgeschlossenen Intervall.

Unter einer *ebenen Kurve* verstehen wir eine stetige Abbildung:

$$\gamma: I \subset \mathbb{R} \rightarrow \mathbb{R}^2, t \mapsto \gamma(t)$$

Die Kurve heisst *glatt*, wenn sie stetig differenzierbar ist. Die Kurve heisst *regulär* sein, wenn der Betrag der Ableitung nicht verschwindet, d.h. $|\dot{\gamma}(t)| \neq 0, \forall t \in I$. Dass die Kurve *geschlossen* sein soll, bedeutet dass $I = [a, b]$ ein (endliches) reelles Intervall ist mit $\gamma(a) = \gamma(b)$, wobei auch die Ableitung übereinstimmt: $\gamma'(a) = \gamma'(b), \forall k \in \mathbb{N}$.

Damit der Rand eines Billardtisches «vernünftig» aussieht, soll die Kurve *einfach geschlossen* sein, das heisst, dass die Abbildung γ auf $[a, b]$ *injektiv* ist. Das heisst auch, dass keine Doppelpunkte auftreten.



Links eine einfach geschlossene Kurve, rechts ist die Kurve nicht einfach geschlossen

Wir werden immer wieder die Ableitung von $|\gamma(t)|$ brauchen. Man kann sich überzeugen, dass gilt:

$$\frac{d}{dt} |\gamma(t)| = \frac{\gamma(t) \cdot \dot{\gamma}(t)}{|\gamma(t)|}$$

Im Zähler steht das Skalarprodukt.

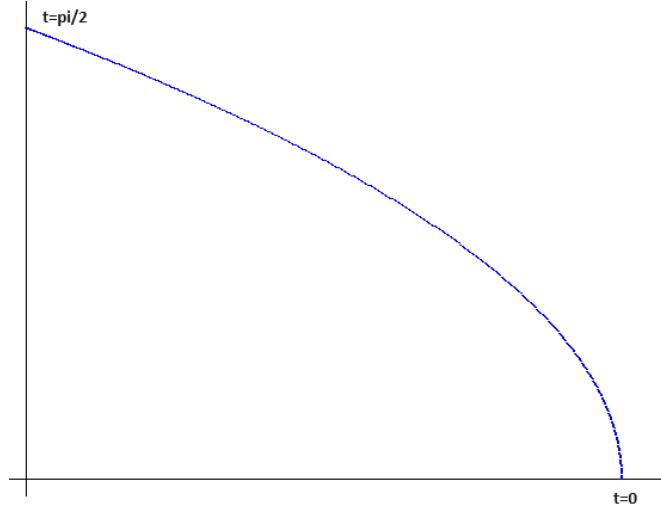
Beispiel

Die ebene Kurve

$$\gamma(t) = \begin{pmatrix} \cos^2 t \\ \sin t \end{pmatrix}, t \in [0, \pi[$$

Ist geschlossen: $\gamma(0) = \gamma(\pi)$. Sie ist glatt, aber nicht regulär: $|\dot{\gamma}(\frac{\pi}{2})| = 0$.

Sie hat «Spitzen» bei $t = 0, \frac{\pi}{2}$.



Graph der oben definierten ebenen Kurve

Im Folgenden betrachten wir nur *konvexe* Billardtische. Das sind Tische, welche von *konvexen* Kurven begrenzt sind. Wir verwenden dabei folgende Definition für die Konvexität:

Eine einfach geschlossene und ebene Kurve heisst konvex, wenn für zwei beliebige Punkte, welche innerhalb des von der Kurve umschlossenen Gebietes liegen, auch deren Verbindungsstrecke ganz in diesem Gebiet liegt.

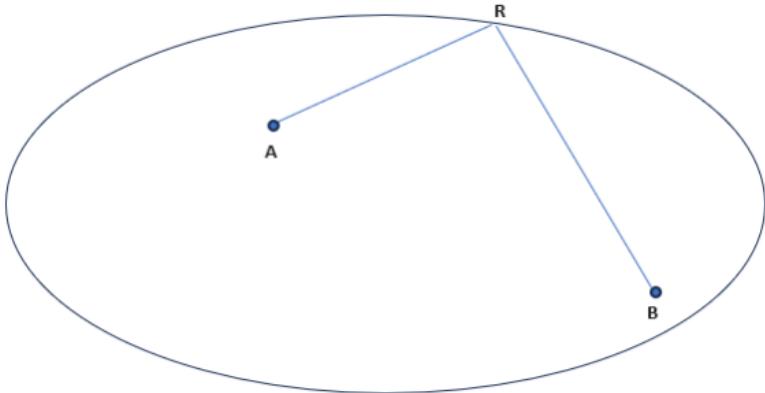
Im Folgenden soll also ein *konvexer Billardtisch* immer von einer ebenen, einfach geschlossenen, glatten, regulären und konvexen Kurve begrenzt werden.

Eine konvexe Kurve kann Geradensegmente enthalten. Wenn sie dies nicht tut, heisst sie *streng konvex*.

Beispiel

Das elliptische und ovale Billard ist streng konvex, das Billard im Stadium ist konvex.

Im Folgenden wollen wir den Begriff des «Reflexionspunktes» definieren. Sei γ der Rand eines konvexen Billardtisches und A, B zwei verschiedene Punkte in seinem Innern. Dann heisst R ein *Reflexionspunkt für A, B* wenn ein von A ausgehender Bahnabschnitt in R so reflektiert wird, dass er nach der Reflexion in B eintrifft.



R ist Reflexionspunkt für A, B

Bemerkung: Beim elliptischen Billard ist jeder Punkt auf der Ellipse ein Reflexionspunkt für die Brennpunkte der Ellipse.

Eine erste Frage, die man stellen kann, ist: *Gibt es für beliebige Punkte $A \neq B$, die im Innern eines konvexen Billardtisches liegen, immer einen Reflexionspunkt R ?*

Wir betrachten die Funktion

$$f(t) = |\overrightarrow{AR}| + |\overrightarrow{RB}|, t \in I$$

Wobei I das Parameterintervall für die Randkurve ist, welche den Billardtisch begrenzt. Sei $\vec{r}(t)$ der zu R gehörende Ortsvektor. Dann ist:

$$f(t) = |\vec{r}(t) - \vec{a}| + |\vec{r}(t) - \vec{b}|$$

f ist eine reellwertige, stetige Funktion und I ein abgeschlossenes Intervall. Wenn f konstant ist, dann sind A und B Brennpunkte einer Ellipse und dann ist jeder Randpunkt ein Reflexionspunkt.

Andernfalls nimmt f auf I ein Maximum und ein Minimum effektiv an. I kann periodisch fortgesetzt werden, da die Randkurve geschlossen ist. Das heisst, dass die Extrema von f im Innern von I oder im Innern der periodischen Fortsetzung auftreten. Weil die Randkurve glatt ist, ist f stetig differenzierbar. Die Ableitung von f an den Extremalstellen ist gleich Null. Seien t_1, t_2 die Parameterwerte, für die f maximal bzw. minimal wird. Dann gilt:

$$\frac{d}{dt} f(t) = \frac{(\vec{r}(t) - \vec{a})}{|\vec{r}(t) - \vec{a}|} \cdot \dot{\vec{r}}(t) + \frac{(\vec{r}(t) - \vec{b})}{|\vec{r}(t) - \vec{b}|} \cdot \dot{\vec{r}}(t) = 0 \text{ für } t = t_{1,2}$$

$\frac{(\vec{r}(t) - \vec{a})}{|\vec{r}(t) - \vec{a}|} =: \vec{e}_a$ ist ein Einheitsvektor und ebenso $\frac{(\vec{r}(t) - \vec{b})}{|\vec{r}(t) - \vec{b}|} =: \vec{e}_b$ in jeweils Richtung des entsprechenden Bahnabschnittes durch den Punkt A bzw. B . Die Summe der beiden Einheitsvektoren $\vec{e}_a + \vec{e}_b$ ist die Winkelhalbierende dieser beiden Bahnabschnitte. Es gilt:

$$(\vec{e}_a + \vec{e}_b) \cdot \dot{\vec{r}}(t_{1,2}) = 0$$

Somit steht diese Winkelhalbierende für $t = t_{1,2}$ senkrecht auf der Tangente in diesen Punkten und damit schliessen die Verbindungsstrecken AR und BR mit der Tangente denselben Winkel ein.

Da der Billardtisch konvex ist, liegen die Verbindungen AR und BR ganz im Innern des Tisches und stellen Bahnabschnitte einer Kugel dar, welche in A startet, in R reflektiert wird und dann auf B stößt.

Als Resultat haben wir:

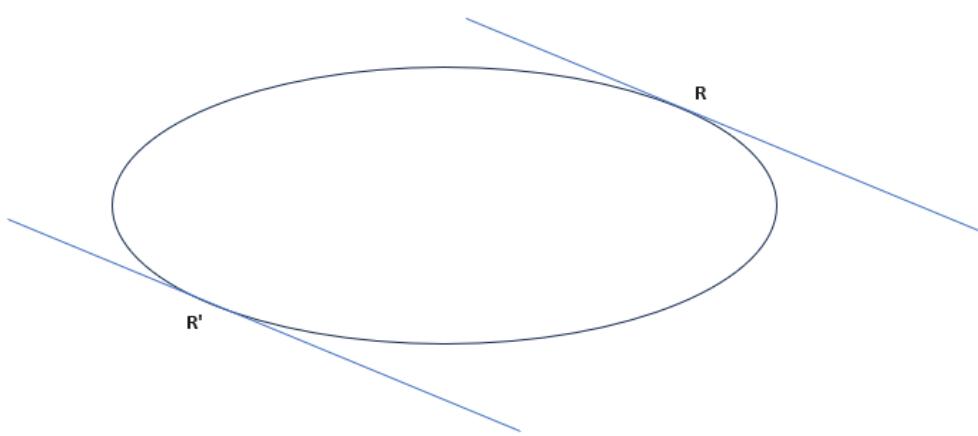
Sind A und B zwei beliebige Punkte im Innern eines konvexen Billardtisches. Dann gibt es mindestens zwei Reflexionspunkte $R_{1,2}$ für A und B . An diesen Stellen hat die Summe der Abstände $|AR| + |BR|$ ein lokales Extremum.

Man kann auch den Fall untersuchen, wo A und B zusammenfallen. Auch dann gilt dasselbe Resultat: Es gibt mindestens zwei Reflexionspunkte, so dass eine von A ausgehende Kugel nach dem Stoß wieder zu A zurückkehrt. Der Nachweis ist eine Übung.

In einigen Beispielen konnten wir Startparameter, welche zu periodischen Bahnen führten, mit einer Intervallschachtelung approximieren. Die Frage ist, ob es für konvexe Billardtische immer periodische Bahnen gibt.

Betrachten wir einen (eventuell nicht konvexen) Billardtisch und eine Tangente an seinen Rand in einem beliebigen Randpunkt. Angenommen, es gibt auf beiden Seiten der Tangente Punkte, die zum Innern des Billardtisches gehören. Dann lassen sich Punkte auf verschiedenen Seiten der Tangente finden, deren Verbindung nicht mehr vollständig im Innern des Billardtisches liegen. Dann wäre der Billardtisch nicht konvex. Wenn umgekehrt der Billardtisch konvex ist und wir eine Tangente in einen beliebigen Randpunkt legen, dann liegen alle Punkte, die zum Billardtisch gehören, auf derselben Seite der Tangente.

Nehmen wir einen konvexen Billardtisch und legen wir die Tangente in einen Randpunkt R . Wenn wir diesen Randpunkt einmal auf dem Tischrand herumlaufen lassen, dann hat sich die Tangente dabei um 2π gedreht. Wegen der Stetigkeit der Ableitung gibt es also einen Randpunkt R' , in welchem die Tangente parallel zur ursprünglichen ist. Wenn wir nun ein solches Tangentenpaar betrachten, ist ihre Distanz die «Breite» des Billardtisches bezüglich dieser Tangenten.



Ein Paar paralleler Tangenten bei einem konvexen Billardtisch

Die Breite des Billardtisches in Abhängigkeit des Randpunktes R oder damit des zugehörigen Parameters t des Ellipsenrandes ist eine periodische, beschränkte reelle Funktion und nimmt somit ein Maximum und ein Minimum an. Diese Funktion kann dargestellt werden als:

$$B(t, t') = |\vec{r}(t) - \vec{r}(t')|$$

An den Extremalstellen sind die partiellen Ableitungen von B nach t und t' gleich Null. Es gilt also:

$$\frac{d}{dt} B(t, t') = \frac{\vec{r}(t) - \vec{r}(t')}{|\vec{r}(t) - \vec{r}(t')|} \cdot \dot{\vec{r}}(t) = 0$$

Somit steht der Vektor $\overrightarrow{RR'}$ senkrecht auf der Tangente im Punkt R . Durch die Ableitung nach t' sieht man, dass dieser Vektor auch auf der Tangente im Punkt R' senkrecht steht. Damit gibt es eine 2-periodische Bahn. Da es mindestens zwei Extremalstellen gibt, folgt:

Bei einem konvexen Billardtisch gibt es mindestens zwei 2-periodische Bahnen. Sie entsprechen der minimalen und maximalen Breite des Tisches, bzw. seinen Durchmessern.

Auf Grund unserer Experimente mit dem «Simulator» kann man vermuten, dass es bei einem konvexen Billardtisch möglich ist, periodische Bahnen jeder Periode zu finden. Das ist tatsächlich so, denn es gilt der

Satz von Birkhoff

Bei einem streng konvexen Billardtisch existieren für jedes teilerfremde paar (p, q) natürlicher Zahlen mit $q \geq 2$ und $p \leq [(q - 1)/2]$ zwei geometrisch verschiedene q -periodische Bahnen mit Umlaufzahl p .

$[(q - 1)/2]$ bezeichnet hier die nächstkleinere natürliche Zahl kleiner als $(q - 1)/2$.

Einen direkten Beweis dieses Satzes findet man in [7]. Dieser Satz ist eigentlich eine Folgerung eines viel allgemeineren Fixpunktsatzes von Poincaré-Birkhoff, der hier nicht erläutert werden kann. Eine gute Einführung mit Anwendung dieses Satzes auf das Billard findet man in [8].

4.12. Das C-Diagramm

Bei unimodalen Funktionen haben wir im sogenannten Feigenbaum-Diagramm die Abhängigkeit der iterierten Funktion von einem Parameter untersucht. Wir haben gesehen, dass für bestimmte Parameterwerte periodisches Verhalten auftrat und für andere Parameterwerte chaotisches Verhalten.

Beim Billard haben wir in all unseren Beispielen einen Parameter C definiert, welcher die Form des Billardtisches bestimmt. Es war:

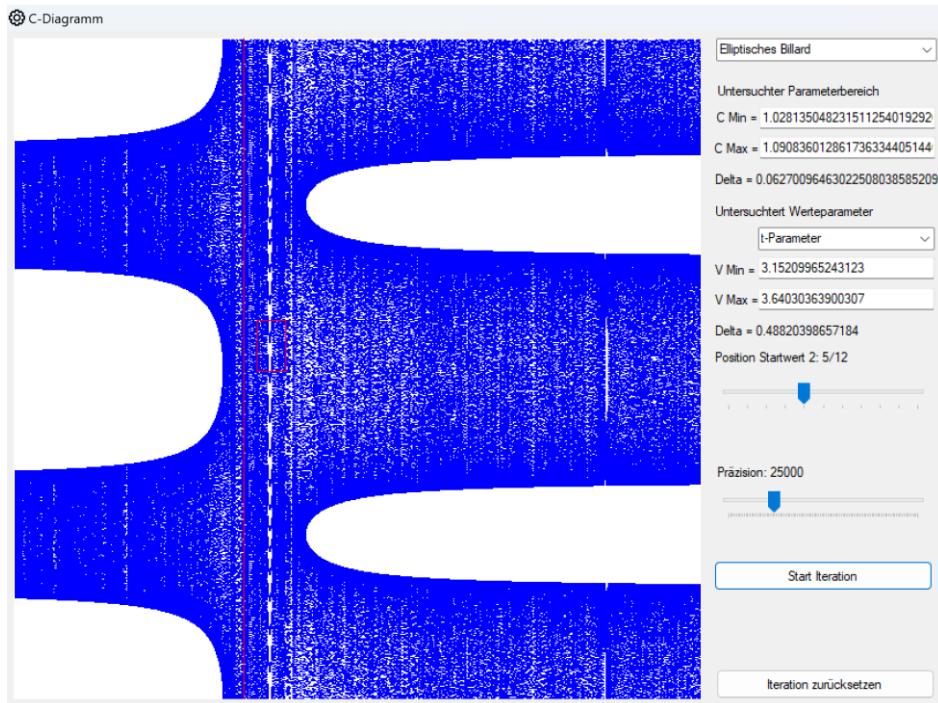
- Elliptisches Billard: $C = \text{Verhältnis der Nebenachse zur Hauptachse der Ellipse}$
- Billard im Stadion: $C = \text{Verhältnis von Kreisdurchmesser zur Breite des Rechtecks im Stadion}$
- Ovales Billard: $C = \text{Verhältnis von Kreisradius zur Hauptachse der Ellipse}$

Beim elliptischen und ovalen Billard ist die Form des Billardtisches für $C = 1$ ein Kreis.

C beeinflusst die Eigenschaften des Billards. Wir haben nun zwei Parameter, um die Bahn einer Kugel auf dem Billardtisch zu beschreiben, nämlich einen Parameter t , welcher den Ort eines Stosspunktes

auf dem Rand des Billardtisches beschreibt und einen Winkel α , welcher den Stosswinkel beschreibt. Das ist der Winkel zwischen Kugelbahn und der Tangente im Stosspunkt.

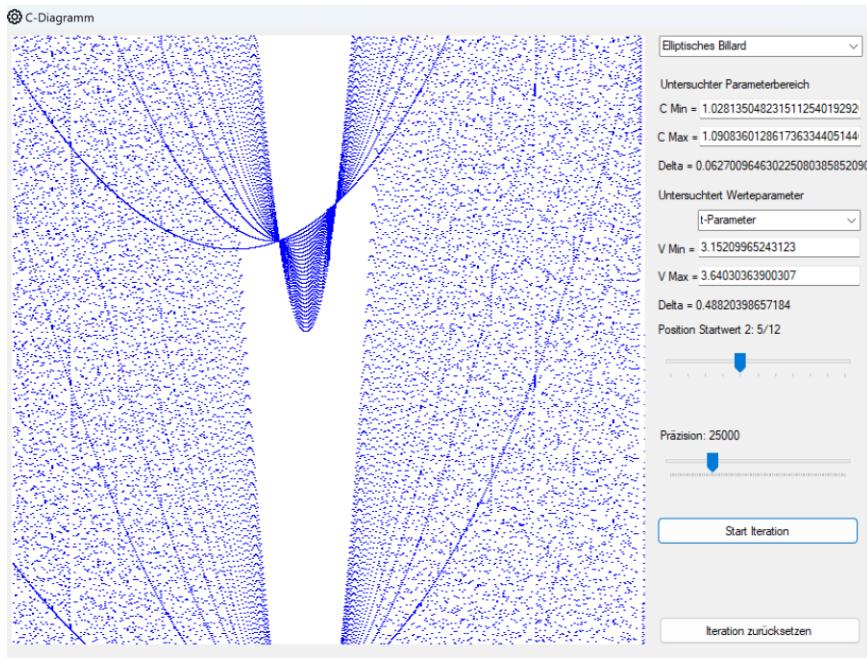
Damit wir den Einfluss von C auf diese Parameter in einem zweidimensionalen Diagramm darstellen können, wählen wir einen der beiden Parameter für die Darstellung aus. Der «Simulator» untersucht dann wie beim Feigenbaum-Diagramm für jeden Wert von C , wie das asymptotische Verhalten des Parameters aussieht.



Abhängigkeit des Parameters t von C beim elliptischen Billard

Die rote vertikale Linie im Diagramm markiert die Stelle $C = 1$.

Wie beim Feigenbaum-Diagramm kann man Ausschnitte aus dem Bild mit gedrückter linker Maustaste markieren. Dann werden die entsprechenden Intervalle für C und für den zu untersuchenden Parameter auf der rechten Seite angepasst. Lässt man die Iteration nochmals laufen, wird der entsprechende Ausschnitt in vergrösserer Form angezeigt:



Vergrösserte Darstellung des früher gewählten Ausschnittes

Für die Iteration treffen wir dabei einige Vereinbarungen:

- C variiert standardmäßig zwischen 0.5 und 2. Diese Werte können rechts im untersuchten Parameterbereich angepasst werden.
- Den Startwert des Parameters, der *nicht* im Diagramm dargestellt werden soll, setzen wir auf $1/3$ seines Parameterintervalls. In den obigen Bildern ist der Winkel $\alpha = \pi/3$.
- Der Startwert des Parameters, der im Diagramm dargestellt wird, kann auf $p/12$ seines Parameterintervalls gesetzt werden, wobei $p \in \{1, 2, \dots, 11\}$. In den obigen Bildern ist $t = 2\pi \cdot \frac{5}{12}$.

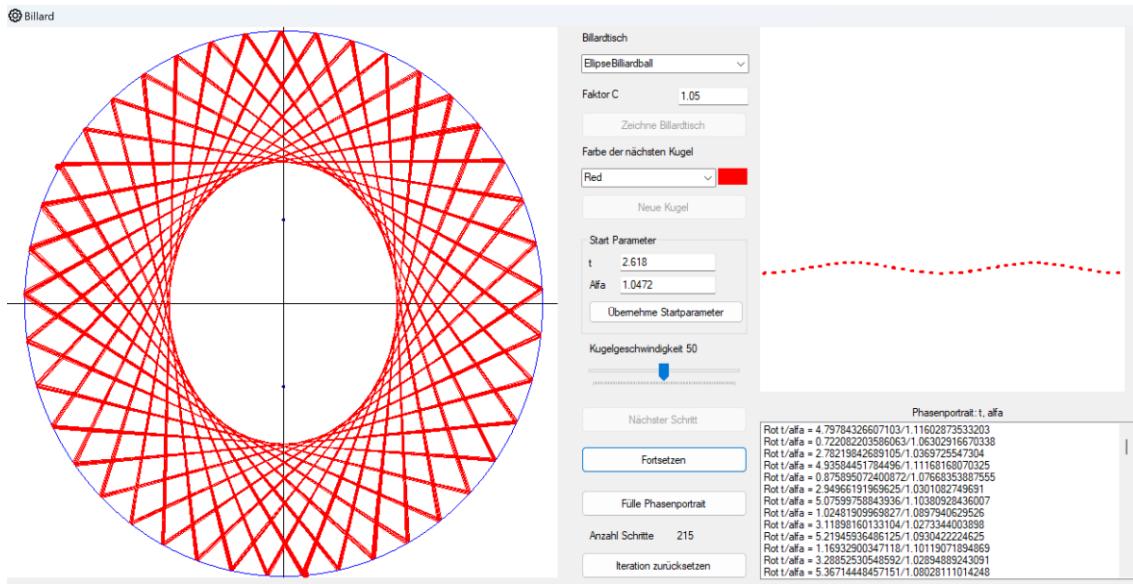
Wenn man das ändern will, kann man leicht den entsprechenden Code in der *FrmCDiagram* und der Subroutine *DrawIterationValues* anpassen.

Es ist wichtig, dass für jedes C für die Iteration wieder *dieselben* Startwerte gesetzt werden, damit die Resultate im obigen Diagramm konsistent und die Vergrösserungen von Ausschnitten korrekt sind.

Im Gegensatz zum Feigenbaum-Diagramm können wir hier leider keine mathematische Analyse der generierten Diagramme anbieten. Immerhin lassen sich aber die Bilder anhand der Form des Billardtisches und der gewählten Startparameter plausibilisieren.

Im vorherigen Diagramm können wir durch ein weiteres Auswahlrechteck abschätzen, für welche Werte von C der in der Mitte des Diagramms sichtbare «Knoten» auftritt. Das ist im Bereich $C \in [1.05, 1.06]$.

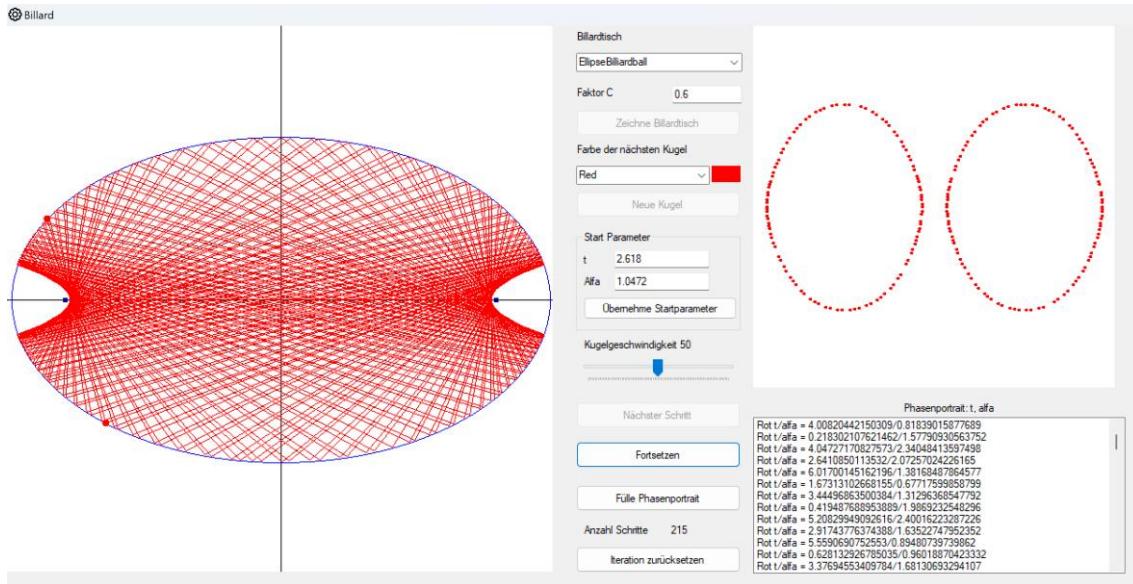
Also gegen wir zum elliptischen Billardtisch, wählen C in diesem Bereich und setzen die Startwerte entsprechend dem C-Diagramm: $t = 2.618, \alpha = 1.0472$.



Die Bahn der Billardkugel mit den vorherigen Parametern

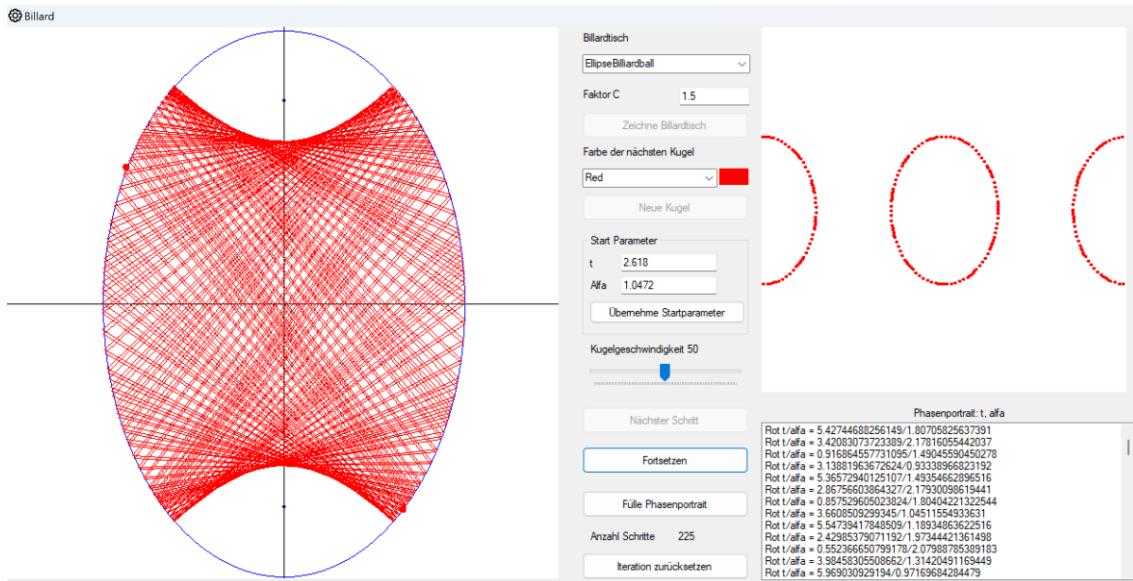
Offenbar sind wir da nahe einer periodischen Bahn mit Periode 40.

Wenn wir nochmals zum ersten Bild zurückkehren, sehen wir, dass für $C = 0.6$ der Parameter t (immer mit denselben Startwerten wie vorhin) in zwei disjunkten Wertintervallen liegt. Das entsprechende Bild auf dem Billardtisch ist:



Die Billardbahn mit den erwähnten Startparametern und $C = 0.6$

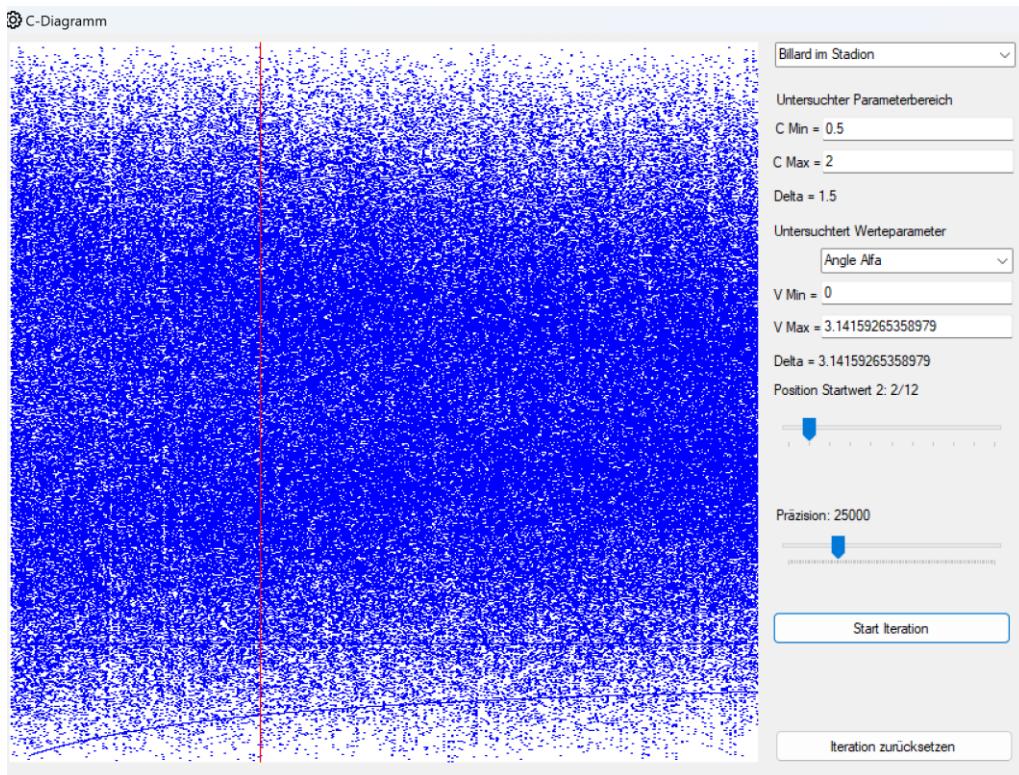
Das Bild im C-Diagramm im Falle $C = 1.5$ ist dreigeteilt: Der Parameter t scheint sich in drei Bereichen zu bewegen. Dabei muss man beachten, dass wegen der Periodizität von t und α das C-Diagramm eigentlich ein Torus ist, indem man den unteren und oberen horizontalen Rand sowie den linken und rechten vertikalen Rand des Diagramms zusammengeklebt. Dann ist auch in diesem Fall der Parameter t in zwei disjunkten Wertintervallen. Das Bild auf dem Billardtisch bestätigt dies.



Der Billardtisch mit den erwähnten Startparametern und $C = 1.5$

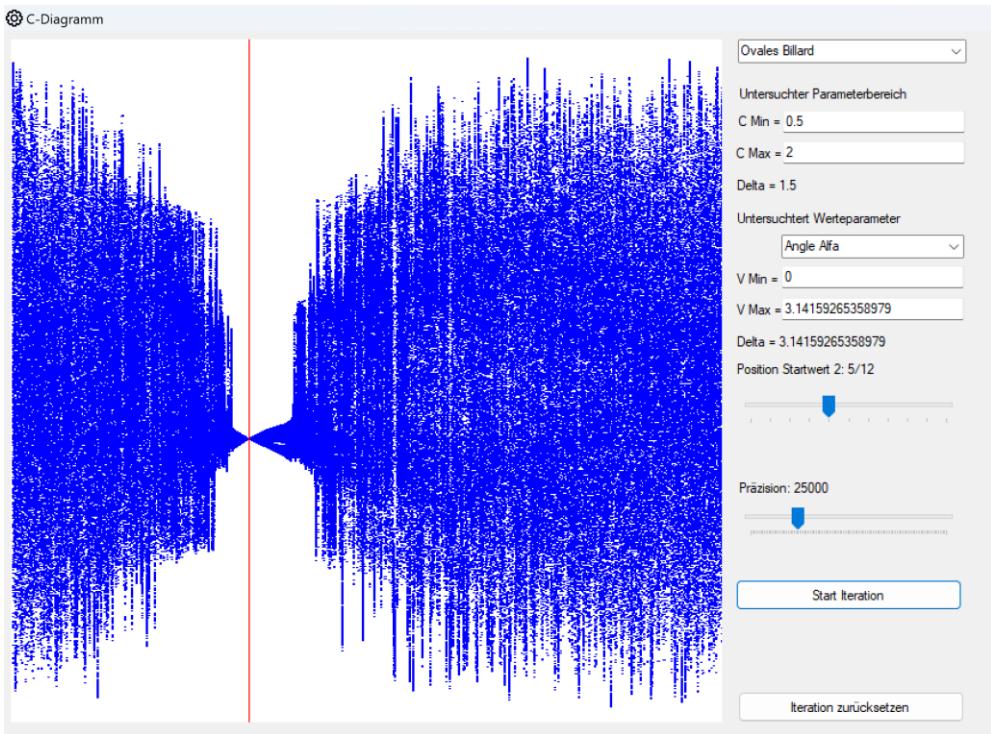
Wir überlassen die weitere Untersuchung am elliptischen Billard dem Leser, insbesondere Diagramme für den Reflexionswinkel α .

Beim Billard im Stadion sieht das C-Diagramm für verschiedene Startwerte immer recht chaotisch aus, wie das folgende Beispiel illustriert.



Billard im Stadion und Diagramm für den Reflexionswinkel α

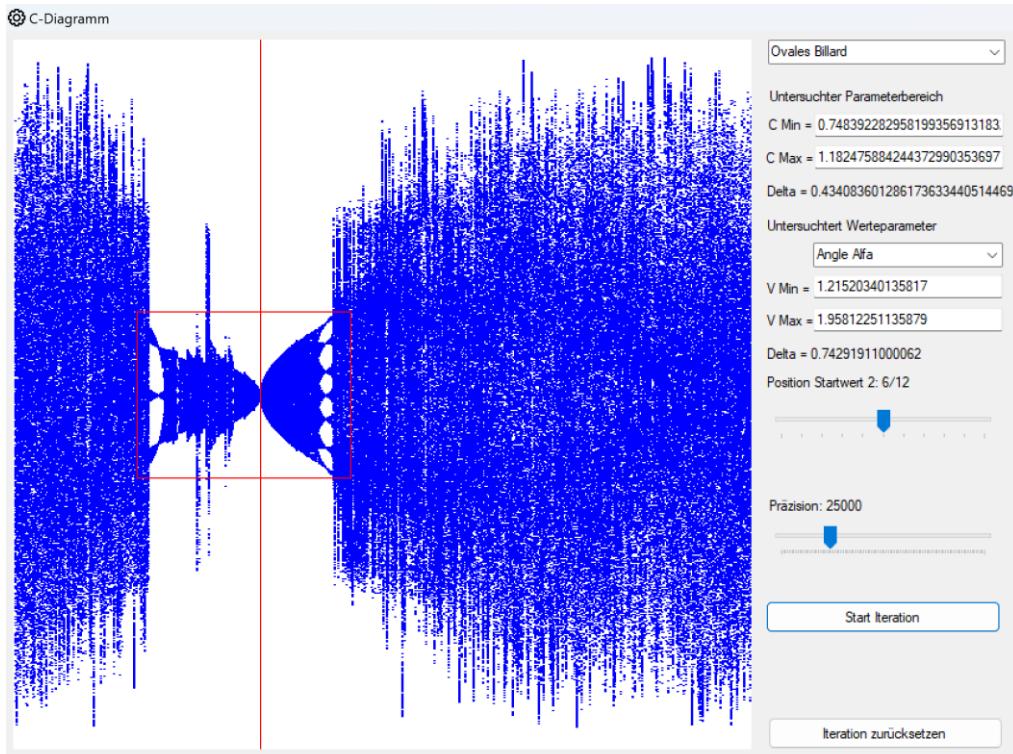
Interessant ist die Betrachtung des ovalen Billards, und zwar für den Reflexionswinkel α .



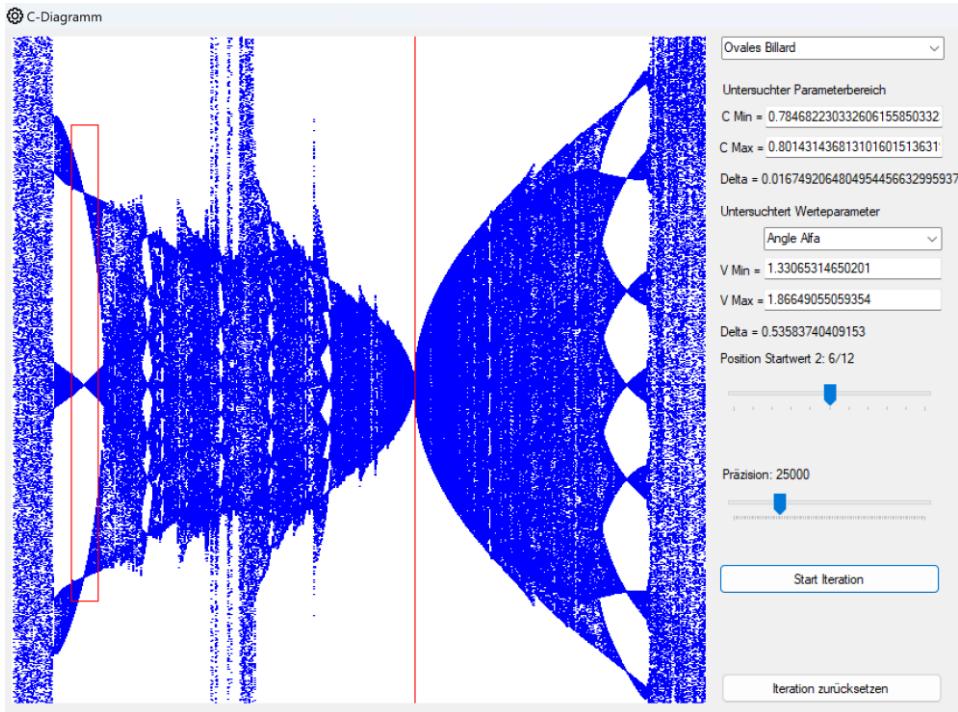
C-Diagramm für das ovale Billard und den Reflexionswinkel α

Im Bereich $C \approx 1$ hat das Diagramm Ähnlichkeit mit dem entsprechenden Diagramm beim elliptischen Billard, wie man selbst mit dem «Simulator» feststellen kann. Ausserhalb dieses Bereichs sieht es eher chaotisch aus.

Ein interessanter Startwert ist $\alpha = \pi \cdot \frac{6}{12}$. Im Bereich $C \approx 1$ sieht man wieder die Ähnlichkeit mit dem elliptischen Billard. Etwas ausserhalb dieses Bereiches zeigt sich aber eine Art periodischer Struktur.



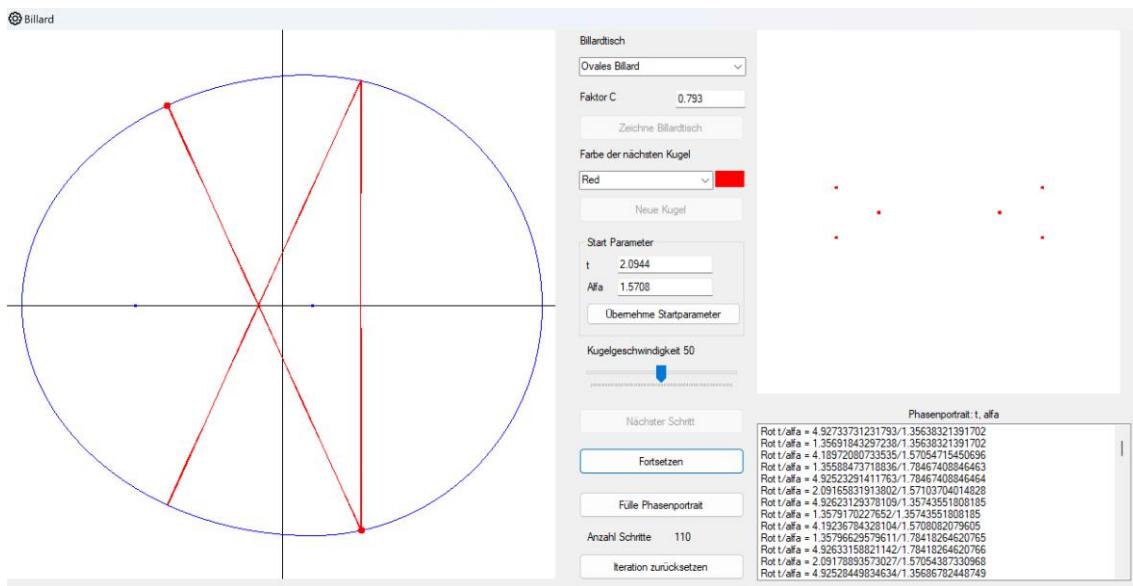
Ein C-Diagramm für das ovale Billard



Der vergrösserte Ausschnitt

Man sieht im obigen Bild durch den weiteren Ausschnitt, dass in der Umgebung von $C \approx 0.793$ eine 3-periodische Bahn aufzutreten scheint.

Betrachten wir den ovalen Billardtisch mit $C = 0.793$ und den Startparametern wie im obigen C-Diagramm: $t = 2.0944, \alpha = 1.5708$.



Bahn der Billardkugel mit obigen Parametern

Offenbar handelt es sich um eine 6-periodische Bahn.

Weitere Untersuchungen überlassen wir dem Leser.

Eine mathematische Analyse der auftretenden Bilder scheint noch offen.

4.13. Übungsbeispiele

1. Untersuche das Billard im Rechteck. Spiegle dazu das Rechteck an allen Seiten und setze diese Spiegelungen an den neuen Rechtecken fort. Diese Technik heisst *Entfaltungsmethode*. Dann wird die Kugelbahn eine Gerade. Welche periodischen Bahnen gibt es? Füllen aperiodische Bahnen das Rechteck dicht aus? Kann man die Bahn durch «Zusammenkleben» geeigneter Seiten des Rechteckgitters als Bahn auf einem Torus interpretieren?
2. Sei $\frac{p}{q}$, $1 \leq p < q$ ein gekürzter Bruch. Gibt es beim Kreisbillard für jeden solchen Bruch eine Kugelbahn mit Periode q und Umlaufzahl p ?
3. Wir haben beim elliptischen Billard durch ein Iterationsverfahren näherungsweise einen Startwinkel α gefunden, welcher ausgehend vom Startpunkt $(a, 0)$ zu einer drei-periodischen Bahn führte. Suche experimentell mit Hilfe des «Simulator» näherungsweise einen Startwinkel, der ausgehend von demselben Startpunkt zu einer 5-periodischen Bahn führt.
4. Die Hyperbel ist definiert als der Ort aller Punkte in der Ebene, so dass der Betrag der Differenz der beiden Abstände von zwei fixen Brennpunkten konstant ist. Diese Konstante bezeichnet man üblicherweise mit $2a$ analog zur Ellipse.

Sei P ein Punkt auf der Hyperbel und F_1, F_2 seien die beiden Brennpunkte. Dann ist sowohl

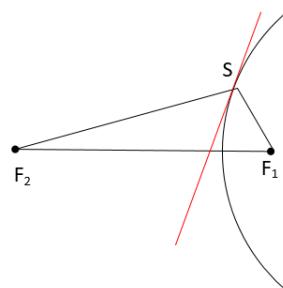
$$|PF_1| - |PF_2| = 2a$$

Wie auch

$$|PF_2| - |PF_1| = 2a$$

Die Hyperbel hat also zwei Zweige.

Anstelle der «Reflexion» eines von einem Brennpunkt ausgehenden Strahles tritt hier die Spiegelung dieses Strahles an der Hyperbeltangente im Treffpunkt des Strahles. Dann geht dieser Strahl nach der Spiegelung durch den anderen Brennpunkt.



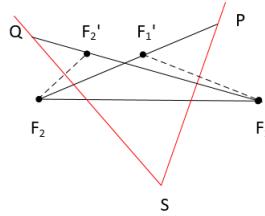
Ein Brennstrahl wird an der Hyperbeltangente (rot) gespiegelt

Das heisst, in obiger Figur schliesst $\overline{SF_1}$ und $\overline{SF_2}$ mit der Hyperbeltangente denselben Winkel ein.

Betrachte nun das elliptische Billard für den Fall, dass die Kugelbahn zwischen den Brennpunkten durchgeht. Zeige, dass in diesem Fall die Kaustik der Bahn eine Hyperbel ist.

Erster Hinweis: Gehe analog vor wie im Fall einer elliptischen Kaustik.

Falls dieser Hinweis nicht zum Ziel führt, hier eine etwas weitergehende Hilfe. Betrachte folgende Figur:



Kaustik für Kugelbahnen, die zwischen den Brennpunkten laufen

Rot dargestellt werden zwei aufeinanderfolgende Kugelbahnen, welche im Punkt S auf den elliptischen Billardtisch treffen. Um die Berührungs punkte der Kugelbahn mit der gesuchten Hyperbel zu finden, kann man analog vorgehen wie wir es bei der Ellipse als Kaustik gemacht haben: Man spiegelt die Brennpunkte am jeweiligen Kugelbahnabschnitt. Die gesuchten Berührungs punkte P und Q liegen dann auf der Verlängerung der Verbindung zwischen F_1F_2' bzw. $F_1'F_2$.

Begründe nun, dass man durch diese Konstruktion wirklich die gesuchten Berührungs punkte erhält, d.h. dass z.B. $\overline{PF_2}$ mit der rechten Kugelbahn denselben Winkel einschließt, wie $\overline{SF_1}$.

Zeige ferner, dass die Dreiecke S, F_1, F_2' und S, F_1', F_2 kongruent sind. Zeige, dass daraus folgt:

$$|F_2'F_1| = |F_1'F_2| \Rightarrow |F_1Q| - |QF_2| = |F_2P| - |PF_1|$$

Somit gehören P und Q zur selben Hyperbel, liegen aber auf dem jeweils anderen Zweig derselben.

Damit haben wir also bewiesen, dass die Kaustik für Kugelbahnen, welche zwischen den Brennpunkten verlaufen, eine Hyperbel ist.

5. Betrachte das Billard im Stadion. Eine Kugel startet im Scheitelpunkt $(0, a + b)$. Gib explizit die Startwinkel $\alpha(k)$ an, für welche die Bahn der Kugel die Periode $2k$, $k \in \mathbb{N}$ besitzt.

6. Betrachte das Billard im Stadion. Eine Kugel startet im Punkt $(0, b)$. Suche durch ein Näherungsverfahren einen Startwinkel α , so dass die Bahn der Kugel die Periode 5 hat.

7. Diskutiere die periodischen Bahnen beim ovalen Billard.

8. Betrachte einen Billardtisch, der von einer konvexen und differenzierbaren Kurve $\gamma(t)$ begrenzt wird. Ferner sei $\gamma(t)$ positiv orientiert, d.h. der Billardtisch liegt immer links der Kurventangente $\dot{\gamma}(t)$. Es sei:

ψ_n = der Winkel zwischen (gerichteter) Kurventangente im n-ten Stosspunkt und positiver x-Achse

α_n = der Reflexionswinkel beim n-ten Stoss

φ_n = der Winkel zwischen dem (gerichteten) n-ten Bahnabschnitt und der positiven x-Achse

Zeige, dass gilt:

$$\begin{cases} \alpha_{n+1} = \psi_{n+1} - \varphi_n \\ \varphi_{n+1} = \psi_{n+1} + \alpha_{n+1} \\ \varphi_{n+1} = 2\psi_{n+1} - \varphi_n \end{cases}$$

9. Sei $\gamma: I \subset \mathbb{R} \rightarrow \mathbb{R}^2, t \mapsto \gamma(t)$ eine reguläre und glatte ebene Kurve. Wenn man die Kurve in Abhängigkeit des Parameters t durchläuft, kann $\dot{\gamma}(t)$ als (tangentialer) Geschwindigkeitsvektor und $\ddot{\gamma}(t)$ als Beschleunigung aufgefasst werden. Diese Beschleunigung kann man zerlegen in einen Vektor parallel und einen senkrecht zum Geschwindigkeitsvektor:

$$\ddot{\gamma}(t) = a^{\parallel}(t) + a^{\perp}(t)$$

Zeige: Wenn γ nach der Bogenlänge s parametrisiert ist, dann gilt: $a^{\parallel}(s) = 0, \forall s$.

Hinweis:

Die *Bogenlänge* einer Kurve ist definiert als:

$$L(\gamma) = \int_a^b |\dot{\gamma}(t)| dt$$

Es vereinfacht viele Rechnungen, wenn man bei einer Kurve die Bogenlänge als Parameter wählt. Das ist im Falle einer regulären Kurve möglich. Wir bezeichnen diesen Parameter mit s und bei dieser Parametrisierung gilt:

$$|\dot{\gamma}(s)| = 1, \forall s \in I$$

10. Die übliche Parameterdarstellung des Kreises mit Radius r ist: $\gamma(t) = r \begin{pmatrix} \cos t \\ \sin t \end{pmatrix}, t \in [0, 2\pi[, r > 0$. Wie sieht eine Parameterdarstellung mit der Bogenlänge als Parameter explizit aus?

11. Sei ein Punkt A im Innern eines konvexen Billardtisches gegeben. Zeige: Dann gibt es mindestens zwei Reflexionspunkte R , so dass eine von A ausgehende und in R reflektierte Kugel wieder zu A zurückkehrt.

12. Analysiere das C-Diagramm für da elliptische Billard und den Reflexionswinkel α . Erkläre die auftretenden Wertebereiche für α anhand der Bahnen auf dem Billardtisch.

13. Analysiere das C-Diagramm für das ovale Billard und untersuche insbesondere auftretende Strukturen, welche periodisch scheinen.

5. Numerische Methoden zur Lösung von gewöhnlichen Differentialgleichungen

Wenn gewöhnliche Differentialgleichungen am Gymnasium behandelt werden, geschieht dies oft in einem ergänzenden Vertiefungsfach. Dabei beschränkt man sich meist auf solche Gleichungen, welche analytisch lösbar sind. Da wir im nächsten Abschnitt dynamische Systeme betrachten werden, welche keine analytische Lösung besitzen, sondern deren Lösung durch numerische Verfahren approximiert wird, soll dieses Kapitel einen Einstieg in dieses Thema bieten. Dabei beschränken wir uns auf einfache Begriffe und Verfahren. Für eine tieferen Einblick in das Thema steht genügend Literatur zur Verfügung. Siehe zum Beispiel [9].

Es werden einige einfache numerische Lösungsverfahren erklärt. Der «Simulator» ermöglicht dann am Beispiel des klassischen Federpendels, diese Lösungen und die jeweils resultierende Pendelbewegung darzustellen. Damit können die Lösungsverfahren untereinander und auch mit der analytischen Lösung verglichen werden.

5.1. Gewöhnliche Differentialgleichungen

In einer Differentialgleichung kommen neben einer gesuchten Funktion auch Ableitungen dieser Funktion vor. Wenn die gesuchte Funktion nur von *einer* Variablen abhängt, dann heißt die Differentialgleichung *gewöhnlich*. Für uns genügt es, gewöhnliche Differentialgleichungen *erster Ordnung* zu betrachten, das heißt, dass nur die *erste* Ableitung der gesuchten Funktion auftritt. Eine solche Differentialgleichung hat die Form:

$$y'(x) = f(x, y(x))$$

Darin ist $y(x)$ eine gesuchte reelle Funktion. Zusätzlich ist eine sogenannte Anfangsbedingung gegeben: $y(a) = y_0$ für $a, y_0 \in \mathbb{R}$.

Oft betrachtet man Funktionen, bei denen die Variable die Zeit t ist. In diesem Fall bezeichnet man die Ableitung gemäß der Newtonschen Notation mit einem Punkt. Die Differentialgleichung erster Ordnung sieht dann so aus:

$$\dot{y}(t) = f(t, y(t))$$

Beispiel

Beim radioaktiven Zerfall hängt die Anzahl der gerade zerfallenden Atome proportional von der Anzahl der vorhandenen Atome ab. Wenn $y(t)$ die Anzahl der Atome im Zeitpunkt t beschreibt, dann ist $\dot{y}(t)$ die Änderung dieser Anzahl zu diesem Zeitpunkt. Diese Änderung ist negativ. Die entsprechende Differentialgleichung lautet:

$$\dot{y}(t) = -\lambda y(t), \lambda > 0$$

Wenn der Zerfall zum Zeitpunkt $t = 0$ startet und am Anfang y_0 Atome vorhanden sind, lautet die Anfangsbedingung: $y(0) = y_0$.

Wie man leicht feststellt, hat diese Differentialgleichung die Lösung:

$$y(t) = y_0 e^{-\lambda t}$$

Es gibt (in der Praxis selten auftretende) Fälle von gewöhnlichen Differentialgleichungen, welche analytische Lösungen besitzen, die mit verschiedenen (oft trickreichen) Methoden ermittelt werden können. Das ist hier aber nicht das Thema.

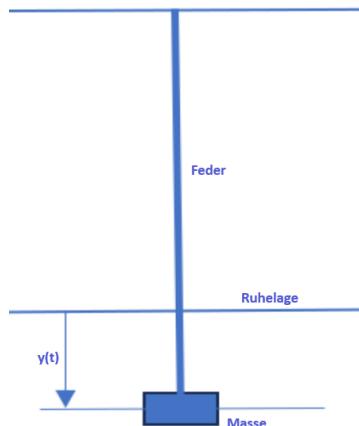
Es gibt drei zentrale Fragen in Zusammenhang mit gewöhnlichen Differentialgleichungen:

- 1) Gibt es eine Lösung? Das heisst, existiert eine Funktion $y(x)$, welche die Differentialgleichung wenigstens in einem Bereich $x \in [a, b]$ erfüllt?
- 2) Ist diese Lösung durch eine Anfangsbedingung $y(a) = y_0$ eindeutig bestimmt?
- 3) Ist die Lösung stabil? Das heisst, dass sich eine kleine Störung der Anfangsbedingung langfristig nur wenig auswirkt.

Wir wollen diese Fragen hier nicht diskutieren, sondern nur darauf hinweisen, dass die Antwort darauf nicht trivial ist. Die dritte Frage ist gerade dann wichtig, wenn die Lösung mit numerischen Methoden nur näherungsweise ermittelt wird.

5.2. Das klassische Federpendel

Wenn wir später numerische Verfahren zur Lösung von gewöhnlichen Differentialgleichungen besprechen, wollen wir diese auf ein Federpendel anwenden und das Ergebnis mit der analytischen Lösung dessen Bewegungsgleichung vergleichen.



Federpendel mit Auslenkung $y(t)$ aus der Ruhelage zum Zeitpunkt t

In Ruhelage heben sich Federkraft und Schwerkraft, welche auf die Masse wirken, auf. Wird die Masse um die Strecke $y(t)$ ausgelenkt, wächst die (rückziehende) Federkraft proportional zu dieser Auslenkung. Der Proportionalitätsfaktor hängt von der Beschaffenheit der Feder ab und wir bezeichnen ihn mit D . Die Schwerkraft ändert sich nicht. Somit wirkt auf die Masse die Federkraft

$$F_{Feder} = -Dy(t)$$

Dadurch wird die Masse beschleunigt. Die Beschleunigung ist $\ddot{y}(t)$ und wenn wir die Masse mit m bezeichnen, gilt nach Newton:

$$m\ddot{y}(t) = -Dy(t)$$

Da wir nur ein idealisiertes Pendel betrachten wollen, setzen wir $m = D = 1$. Damit haben wir die Differentialgleichung:

$$\ddot{y}(t) + y(t) = 0$$

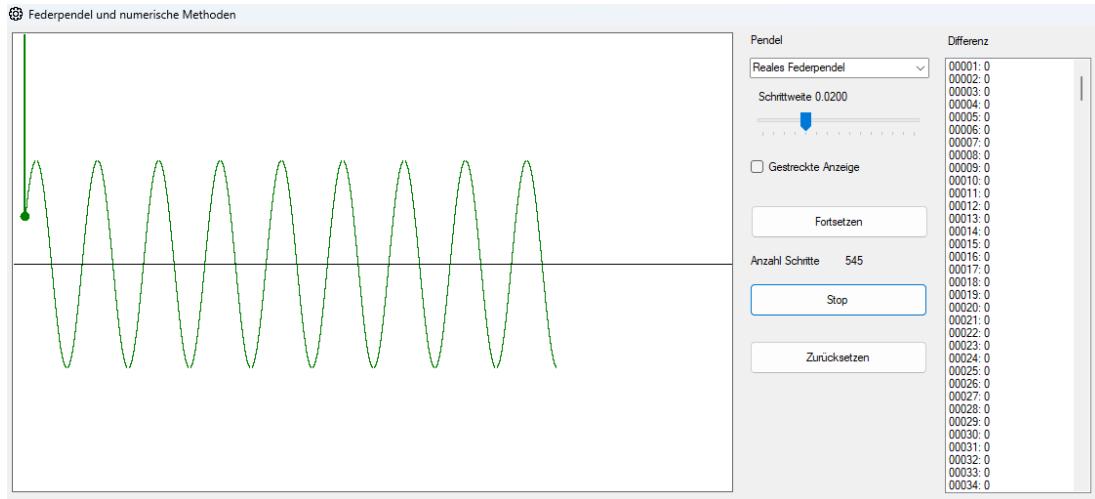
Wir nehmen ferner an, dass das Pendel beim Start um die Strecke y_0 aus der Ruhelage ausgelenkt und dann losgelassen wird. Es gilt also die Anfangsbedingung: $y(0) = y_0$.

Wie man leicht überprüft, ist

$$y(t) = y_0 \cos(t)$$

Eine Lösung dieser Differentialgleichung und diese ist eindeutig bestimmt (ohne dass wir dies hier näher begründen).

Im «Simulator» wird man das Federpendel mit der Maus in die Startposition ziehen und dann sieht man, wie die Schwingung aussieht:



Schwingung eines Federpendels

Dabei ist die Zeitachse etwas gestaucht, da sie von der Geschwindigkeit abhängt, mit welcher das Diagramm in der Implementierung nach rechts versetzt wird. Das gilt aber für alle Pendel in gleichem Masse und beeinflusst den Vergleich verschiedener Methoden nicht.

5.3. Das explizite Euler Verfahren

Wenn die Differentialgleichung keine analytische Lösung besitzt, versucht man, die gesuchte Funktion schrittweise und approximativ zu finden, indem man den Differentialquotienten $y'(x) = \frac{dy}{dx}$ durch den Differenzenquotienten ersetzt: $\frac{dy}{dx} \approx \frac{\Delta y}{\Delta x}$ wobei man Δx klein genug wählt. Dann wird gesetzt:

$$\frac{\Delta y}{\Delta x} = f(x, y(x)) \text{ bzw. } \Delta y = f(x, y(x))\Delta x$$

Das explizite Euler Verfahren (auf Englisch «Forward Euler») arbeitet nun folgendermassen:

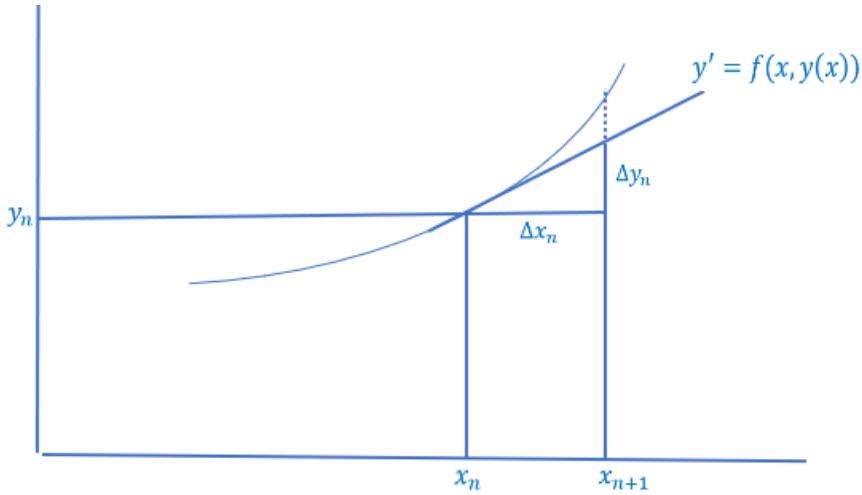
Ausgehend von einer Anfangsbedingung

$$\begin{cases} x_1 = 0 \\ y_1 = y_0 \end{cases}$$

Ermittelt man iterativ weitere Werte von x und $y(x)$ durch die Rekursionsformel:

$$\begin{cases} x_{n+1} = x_n + \Delta x_n \\ y_{n+1} = y_n + \Delta y_n = y_n + f(x_n, y_n)\Delta x_n \end{cases}$$

Das heisst, wir verwenden zur Approximation des nächstens Funktionswert eine lineare Näherung mit Hilfe der Tangentensteigung im Punkt (x_n, y_n) .



Skizze zur expliziten Euler Methode

Im allgemeinen Fall kann die Schrittweite Δx_n bei jedem Schritt variieren. Sie könnte an die erwartete Form der Funktion angepasst werden und dafür gibt es verschiedene Methoden. Man spricht dann von einem *adaptiven* Approximationsverfahren.

Für unsere Zwecke werden wir aber die Schrittweite konstant wählen: $\Delta x_n := h, \forall n$ für ein kleines positives h . Damit wird die Rekursionsformel:

$$\begin{cases} x_{n+1} = x_n + h \\ y_{n+1} = y_n + \Delta y_n = y_n + f(x_n, y_n)h \end{cases}$$

Dieses Verfahren heisst *explizit*, weil die bei jedem Iterationsschritt zu bestimmenden Größen (x_{n+1}, y_{n+1}) bereits auf der linken Seite der Gleichungen stehen.

Nun wollen wir dieses Verfahren auf das Federpendel anwenden. Die entsprechende Differentialgleichung lautete:

$$\ddot{y}(t) + y(t) = 0$$

Das ist eine Differentialgleichung zweiter Ordnung, aber wir können sie in ein Differentialgleichungssystem erster Ordnung umwandeln, indem wir substituieren:

$$\begin{cases} u(t) := y(t) \\ v(t) := y'(t) \end{cases}$$

Das führt zu folgendem Differentialgleichungssystem:

$$\begin{cases} u'(t) := v(t) \\ v'(t) := -u(t) \end{cases}$$

Die frühere Funktion $f(x, y(x))$ besteht also hier aus zwei Komponenten:

$$\begin{cases} f_1(t, u(t), v(t)) = v(t) \\ f_2(t, u(t), v(t)) = -u(t) \end{cases}$$

Die Anfangsbedingungen sind:

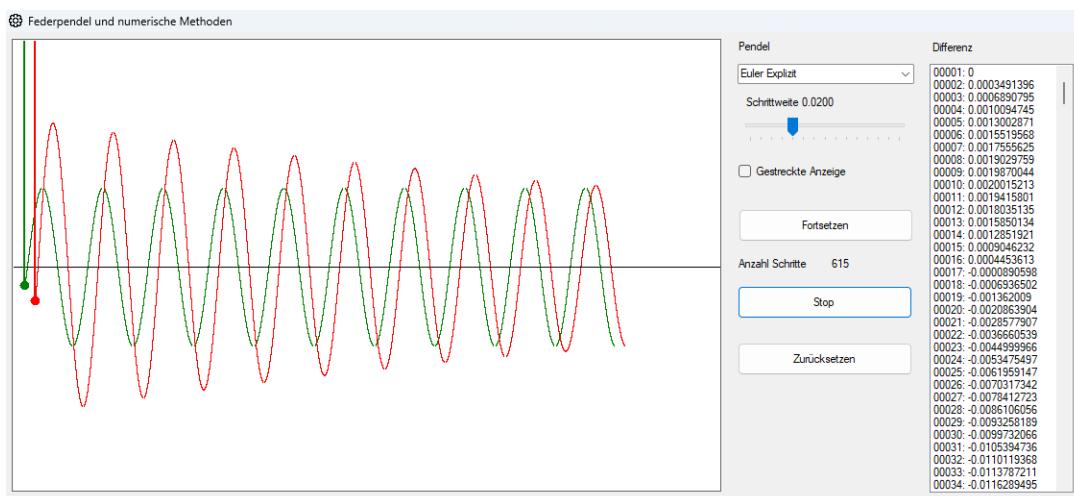
$$\begin{cases} t_1 = 0 \\ u_1 = y_0 \\ v_1 = 0 \end{cases}$$

Es ist $v_1 = 0$, weil das Pendel in der Ausgangsposition keine Geschwindigkeit aufweist.

Nun wählen ein kleines $h > 0$ als konstante Schrittweite für den Zuwachs von t und wenden das explizite Euler Verfahren auf die Komponenten $u(t), v(t)$ an. Das liefert die Rekursionsformeln:

$$\begin{cases} t_{n+1} = t_n + h \\ u_{n+1} = u_n + v_n h \\ v_{n+1} = v_n - u_n h \end{cases}$$

Mit dem «Simulator» können wir nun prüfen, wie sich dieses Verfahren gegenüber der analytischen Lösung für das Federpendel verhält. Dabei können wir die Schrittweite im «Simulator» anpassen. Details zur Implementation folgen in einem späteren Abschnitt.



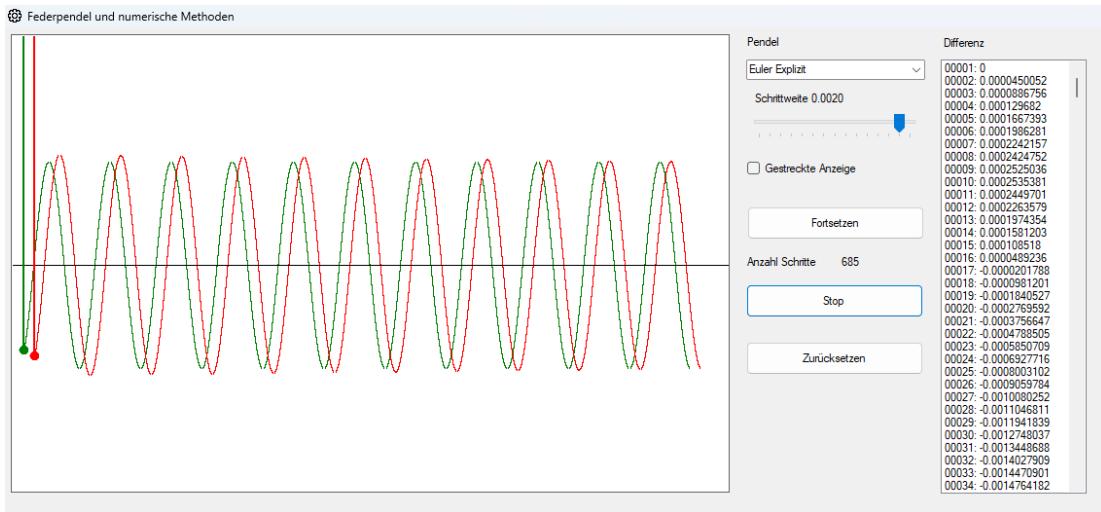
Vergleich der analytischen Lösung mit «Euler explizit»

Im obigen Bild ist das Pendel, welches gemäss der analytischen Lösung schwingt, grün dargestellt. Rot ist die numerische Approximation gemäss dem expliziten Euler Verfahren. Das Verfahren kann oben rechts gewählt werden. Auf der rechten Seite wird zudem die Differenz der beiden Schwingungen bei jedem Iterationsschritt aufgelistet. Beim Start werden beide Pendel mit gedrückter linker Maustaste in dieselbe Startposition gebracht.

Wie man sieht, ist die numerische Approximation hinsichtlich Schwingungsfrequenz recht gut, aber die Amplitude läuft «aus dem Ruder».

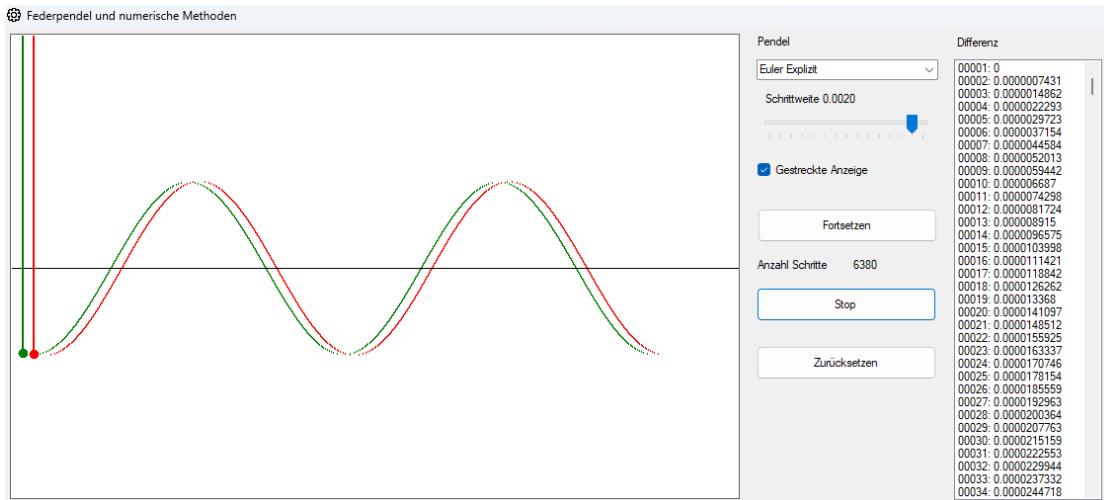
Wir können die Schrittweite hinunter setzen. Die Schrittweite für das grüne Pendel ist konstant (im «Simulator» gleich 0.1). Für das rote Pendel sind dann kleinere Schrittweiten der Form $\frac{0.1}{k}$, $k \in \mathbb{N}$ möglich. Das heisst, das rote Pendel schwingt k -Mal, bis seine Position aufgezeichnet wird. Damit sind die Pendel synchron.

Nachfolgend ein Beispiel mit der Schrittweite 0.002 für das rote Pendel. Der Effekt der ausufernden Amplitude ist weiterhin vorhanden. Es geht aber länger, bis er sich auswirkt.



Das rote Pendel schwingt mit einer kleineren Schrittweite

Der «Simulator» ermöglicht auch, dass *beide* Pendel mit derselben kleineren Schrittweite schwingen. Dazu wird die Option *Gestreckte Anzeige* aktiviert. Das liefert eine Art «Zoom» auf das Schwingungsbild:



Beide Pendel schwingen mit derselben Schrittweite

5.4. Das implizite Euler Verfahren

Beim expliziten Euler Verfahren hatten wir für die Approximation die Rekursionsformel:

$$\begin{cases} x_{n+1} = x_n + h \\ y_{n+1} = y_n + \Delta y_n = y_n + f(x_n, y_n)h \end{cases}$$

Beim impliziten Euler Verfahren (auf Englisch «Backward Euler») verwendet man für die Näherung nicht die Tangentensteigung im Punkt (x_n, y_n) , sondern jene im zu bestimmenden Punkt (x_{n+1}, y_{n+1}) . Die Iterationsformal lautet also:

$$\begin{cases} x_{n+1} = x_n + h \\ y_{n+1} = y_n + \Delta y_n = y_n + f(x_{n+1}, y_{n+1})h \end{cases}$$

Das heisst, dass die gesuchten Variablen *implizit* im Gleichungssystem stecken, welches nach diesen Variablen aufgelöst werden muss. Dabei ist es nicht garantiert, dass dies überhaupt möglich ist. Wir

wollen aber nur auf diese Problematik hinweisen. Im Falle des Federpendels gelingt dies, wie wir sehen werden.

Das zu lösende Differentialgleichungssystem ist nach wie vor dasselbe:

$$\begin{cases} u'(t) := v(t) \\ v'(t) := -u(t) \end{cases}$$

Aber an Stelle der Rekursionsformeln

$$\begin{cases} t_{n+1} = t_n + h \\ u_{n+1} = u_n + v_n h \\ v_{n+1} = v_n - u_n h \end{cases}$$

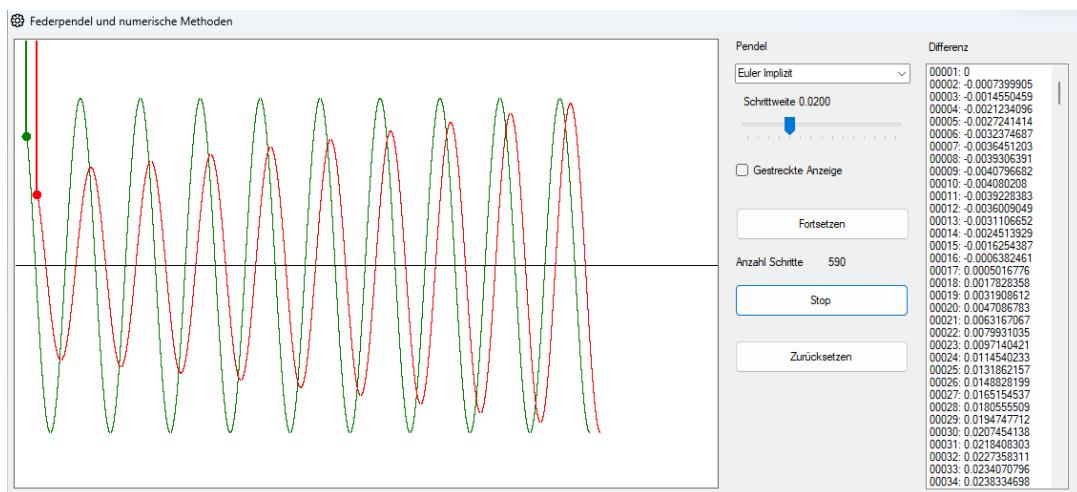
des expliziten Verfahrens, haben wir beim impliziten Verfahren die Formeln:

$$\begin{cases} t_{n+1} = t_n + h \\ u_{n+1} = u_n + v_{n+1} h \\ v_{n+1} = v_n - u_{n+1} h \end{cases}$$

Wenn man dieses Gleichungssystem nach den gesuchten Variablen (u_{n+1}, v_{n+1}) auflöst, erhält man (wie man leicht nachrechnen kann) die Formeln:

$$\begin{cases} t_{n+1} = t_n + h \\ u_{n+1} = (u_n + v_n h) / (1 + h^2) \\ v_{n+1} = (v_n - u_n h) / (1 + h^2) \end{cases}$$

Der Divisor $(1 + h^2) > 1$ wirkt dämpfend auf die Amplitude. Das bestätigt sich, wenn wir das Resultat im «Simulator» anschauen.



Das implizite Euler Verfahren angewandt auf das Federpendel

5.5. Die implizite Mittelpunktregel

Beim expliziten Euler Verfahren ist die Amplitude zu stark angewachsen. Beim impliziten Verfahren wurde sie zu stark gedämpft. Die Frage ist, ob man «etwas in der Mitte» beider Verfahren finden kann.

Das heisst, dass man weder die Tangentensteigung im Punkt (x_n, y_n) noch im Punkt (x_{n+1}, y_{n+1}) für die Approximation verwendet, sondern die Tangentensteigung im Punkt $(x_n + \frac{h}{2}, y_n + \frac{\Delta y_n}{2})$.

Wir bezeichnen: $\hat{y}_n := y_n + \frac{\Delta y_n}{2}$.

Dann lautet die Gleichung zum Bestimmen von \hat{y}_n :

$$\hat{y}_n = y_n + \frac{h}{2} f(x_n + \frac{h}{2}, \hat{y}_n)$$

das heisst, wir verwenden dazu die Tangentensteigung im Punkt $(x_n + \frac{h}{2}, \hat{y}_n)$.

Mit diesem Wert berechnen wir dann y_{n+1} unter Verwendung wieder dieser Tangentensteigung:

$$y_{n+1} = y_n + h f(x_n + \frac{h}{2}, \hat{y}_n)$$

Wie man sieht, handelt es sich um ein implizites Verfahren, weil die erste Gleichung nach \hat{y}_n aufgelöst werden muss.

Nun wenden wir dieses Verfahren auf das Federpendel an. Ausgangspunkt ist wieder das Differentialgleichungssystem:

$$\begin{cases} u'(t) := v(t) \\ v'(t) := -u(t) \end{cases}$$

Zuerst berechnen wir (\hat{u}_n, \hat{v}_n) gemäss der obigen Formel:

$$\begin{cases} \hat{u}_n = u_n + \frac{h}{2} f_1 \left(t + \frac{h}{2}, \hat{u}_n, \hat{v}_n \right) = u_n + \frac{h}{2} \hat{v}_n \\ \hat{v}_n = v_n + \frac{h}{2} f_2 \left(t + \frac{h}{2}, \hat{u}_n, \hat{v}_n \right) = v_n - \frac{h}{2} \hat{u}_n \end{cases}$$

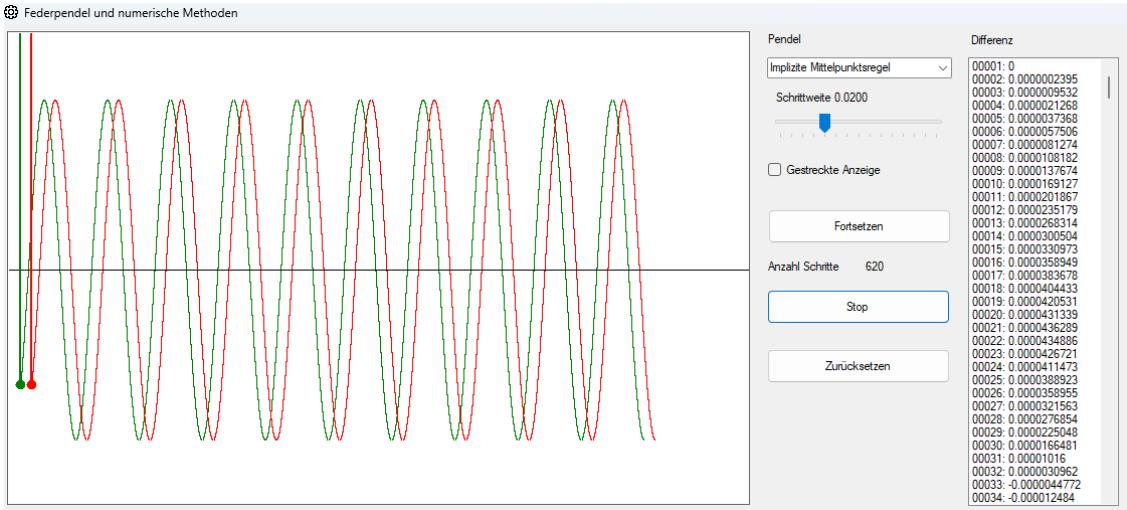
Löst man das Gleichungssystem nach (\hat{u}_n, \hat{v}_n) auf, erhält man:

$$\begin{cases} \hat{u}_n = (u_n + \frac{h}{2} v_n) / (1 + h^2/4) \\ \hat{v}_n = (v_n - \frac{h}{2} u_n) / (1 + h^2/4) \end{cases}$$

Mit diesen Werten berechnen wir (u_{n+1}, v_{n+1}) :

$$\begin{cases} u_{n+1} = u_n + h f_1 \left(t + \frac{h}{2}, \hat{u}_n, \hat{v}_n \right) = u_n + h \hat{v}_n = u_n + h(v_n - \frac{h}{2} u_n) / (1 + h^2/4) \\ v_{n+1} = v_n + h f_2 \left(t + \frac{h}{2}, \hat{u}_n, \hat{v}_n \right) = v_n - h \hat{u}_n = v_n - h(u_n + \frac{h}{2} v_n) / (1 + h^2/4) \end{cases}$$

Im «Simulator» ist diese Rekursion implementiert und jetzt sind wir gespannt auf das Resultat:



Das rote Pendel wird durch die implizite Mittelpunktregel approximiert

Wie das Bild zeigt, sieht das schon ganz ordentlich aus. Das hat aber nicht damit zu tun, dass die implizite Mittelpunktregel besonders gut und für praktische Anwendungen geeignet ist. Es ist einfach so, dass diese Regel im Falle des Federpendels oder ähnlichen Problemen eine gute Approximation liefert. Wie man rechts in der Liste der Differenzen sieht, sind diese aber nicht Null.

5.6. Das Runge-Kutta Verfahren vierter Ordnung

Bei der impliziten Mittelpunktregel haben wir die Tangentensteigung in einem Punkt zwischen (x_n, y_n) und (x_{n+1}, y_{n+1}) für die Approximation verwendet. Eine Idee ist nun, die Tangentensteigung in verschiedenen Punkten in der Nähe von (x_n, y_n) und (x_{n+1}, y_{n+1}) zu berechnen und dann für die Approximation ein gewichtetes Mittel dieser Tangentensteigungen zu verwenden. Diese Idee verwenden die Runge-Kutta Verfahren, benannt nach ihren Entwicklern Carl Runge und Martin Wilhelm Kutta Anfang des 20. Jahrhunderts.

Wie man zur Auswahl der Stützstellen für die Tangentensteigungen und zum gewichteten Mittel derselben kommt, findet man in der Literatur, zum Beispiel in [9]. Wir wollen hier nur das Verfahren zeigen, und zwar eines der Ordnung vier.

Ausgangspunkt ist wie immer eine Differentialgleichung der Form:

$$y'(x) = f(x, y(x))$$

und eine Rekursionsformel beim vierstufigen Verfahren:

$$y_{n+1} = y_n + h \sum_{j=1}^4 b_j k_j$$

Darin sind k_j die Tangentensteigungen in gewissen Punkten und b_j die Koeffizienten des gewichteten Mittels, das heisst dass gilt: $\sum_{j=1}^4 b_j = 1$.

Die Koeffizienten k_j für die Tangentensteigungen sind nun definiert als:

$$k_1 = f(x_n, y_n)$$

$$k_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} k_1\right)$$

$$k_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right)$$

$$k_4 = f(x_n + h, y_n + hk_3)$$

Damit bildet man das gewichtete Mittel für die Approximation, welche lautet:

$$\begin{cases} x_{n+1} = x_n + h \\ y_{n+1} = y_n + h(k_1 + 2k_2 + 2k_3 + k_4)/6 \end{cases}$$

Nun wenden wir das wieder auf das Federpendel an. Das entsprechende Differentialgleichungssystem war gegeben durch:

$$\begin{cases} u'(t) = f_1(t, u(t), v(t)) = v(t) \\ v'(t) = f_2(t, u(t), v(t)) = -u(t) \end{cases}$$

Da wir zwei Komponenten haben, brauchen wir zwei Gruppen von Koeffizienten für die Tangentensteigungen. Wir bezeichnen sie mit k_j für die erste Komponente und l_j für die zweite.

Damit haben wir zuerst:

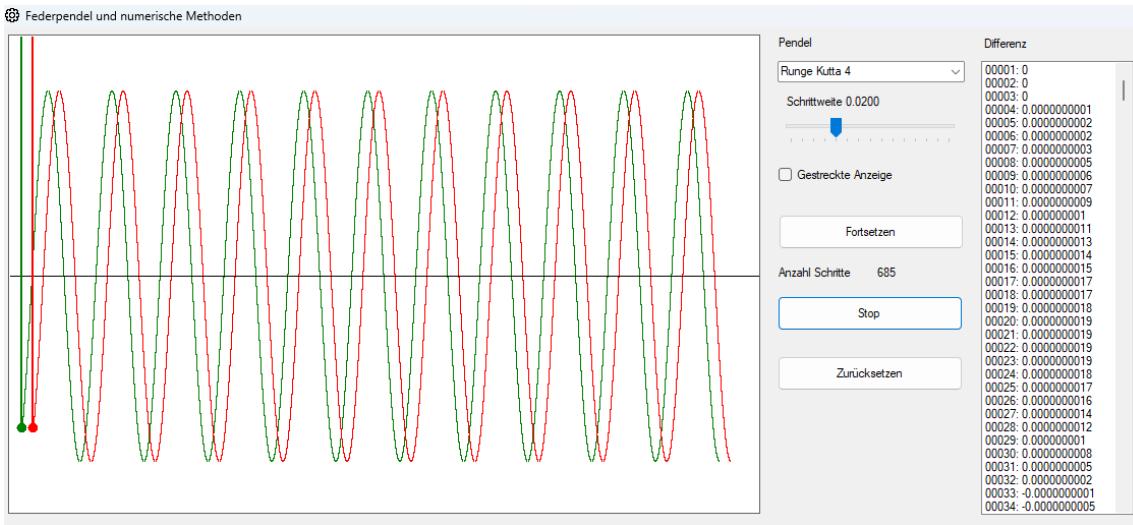
$$\begin{aligned} & \begin{cases} k_1 = f_1(t_n, u_n, v_n) = v_n \\ l_1 = f_2(t_n, u_n, v_n) = -u_n \end{cases} \\ & \begin{cases} k_2 = f_1\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}k_1, v_n + \frac{h}{2}l_1\right) = v_n + \frac{h}{2}l_1 \\ l_2 = f_2\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}k_1, v_n + \frac{h}{2}l_1\right) = -u_n - \frac{h}{2}k_1 \end{cases} \\ & \begin{cases} k_3 = f_1\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}k_2, v_n + \frac{h}{2}l_2\right) = v_n + \frac{h}{2}l_2 \\ l_3 = f_2\left(t_n + \frac{h}{2}, u_n + \frac{h}{2}k_2, v_n + \frac{h}{2}l_2\right) = -u_n - \frac{h}{2}k_2 \end{cases} \\ & \begin{cases} k_4 = f_1(t_n + h, u_n + hk_3, v_n + hl_3) = v_n + hl_3 \\ l_4 = f_2(t_n + h, u_n + hk_3, v_n + hl_3) = -u_n - hk_3 \end{cases} \end{aligned}$$

Damit erhält man die Rekursionsformeln:

$$\begin{cases} t_{n+1} = t_n + h \\ u_{n+1} = u_n + h(k_1 + 2k_2 + 2k_3 + k_4)/6 \\ v_{n+1} = v_n + h(l_1 + 2l_2 + 2l_3 + l_4)/6 \end{cases}$$

Wie man sieht, ist dies ein explizites Verfahren.

Im «Simulator» ist es implementiert. Man erhält damit folgendes Resultat:



Das rote Pendel wird durch das Runge-Kutta Verfahren der Ordnung vier approximiert

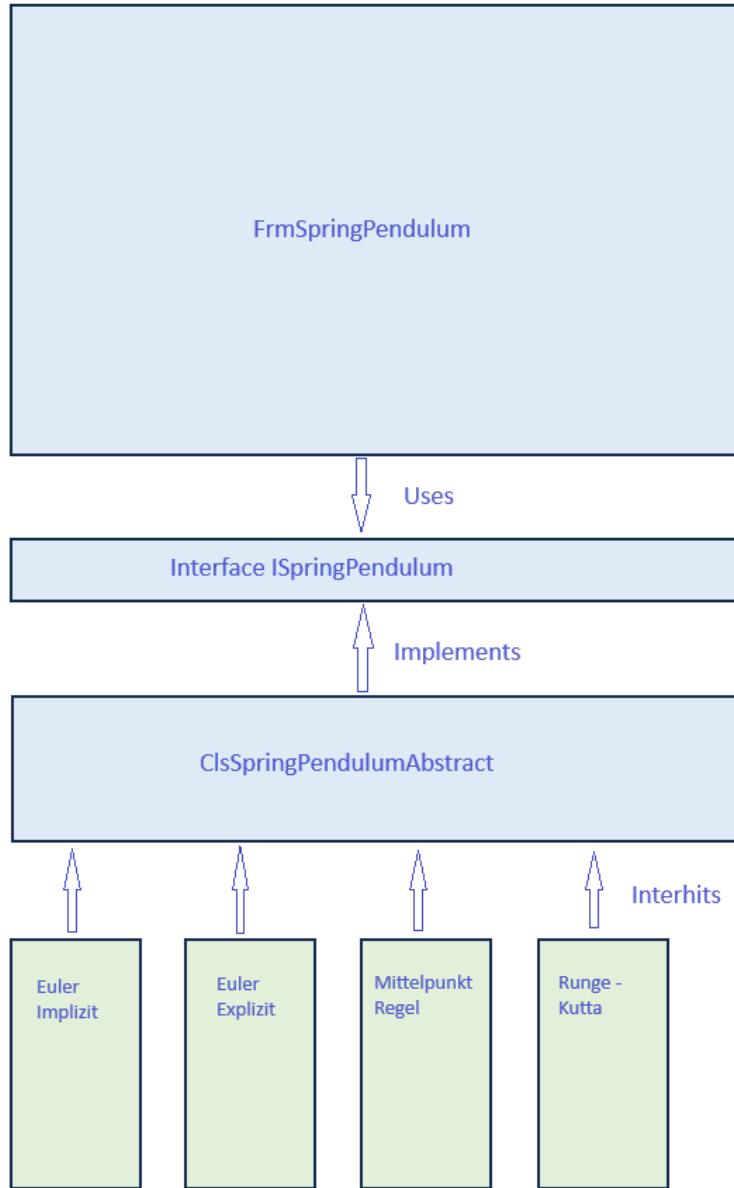
Auch diese Approximation ist optisch recht gut. Auf der rechten Seite sieht man in der Liste der Differenzen, dass diese deutlich kleiner geworden sind.

Wir werden im nächsten Kapitel für die Simulation von verschiedenen Arten von Pendeln dieses Runge-Kutta Verfahren für die Approximation verwenden, weil es einfach zu implementieren und mindestens von der Idee her elementar verständlich ist. Es hat eine gute Balance zwischen Genauigkeit und Rechenaufwand. Es benötigt keine höheren Ableitungen der Funktion als die erste. Zudem ist der Fehler proportional zur fünften Potenz der Schrittweite, was für unsere Zwecke genügt.

In der Praxis gibt es noch viel effizientere Verfahren, teils Mehrschrittverfahren oder adaptive Verfahren mit angepasster Schrittweite pro Schritt. Wer sich dafür interessiert, sei auf die umfangreiche Literatur verwiesen. Eine ganze Palette von praxistauglichen Verfahren, welche oft von Ingenieuren genutzt werden, bietet auch MATLAB mit seiner ODE (= Ordinary Differential Equation) Suite.

5.7. Implementierung im «Simulator»

Die Architektur des Bereichs «Numerische Methoden» ist nach denselben Prinzipien aufgebaut, wie die vorhergehenden Implementierungen: Für den Benutzer soll es einfach möglich sein, seine eigenen numerischen Methoden zu implementieren und auszuprobieren, ohne dass der bestehende Code dadurch gestört wird. Der Aufbau ist folgender:



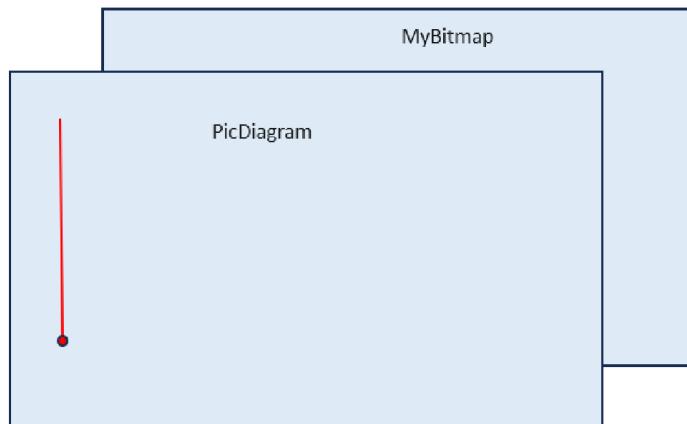
Architektur bei der Implementierung der Federpendel

Der Benutzer kann seine eigene numerische Methode ausprobieren, indem er eine weitere Klasse mit dieser Methode schreibt und von der abstrakten Klasse *ClsSpringPendulumAbstract* erbt. Dabei muss er lediglich die Routine *Protected MustOverride Sub Iteration* dieser Klasse implementieren durch eine Routine *Protected Overrides Sub Iteration*. Darin befinden sich dann die Details zum numerischen Verfahren.

Tricky ist allenfalls das Schieben der Pendelbahn nach rechts.

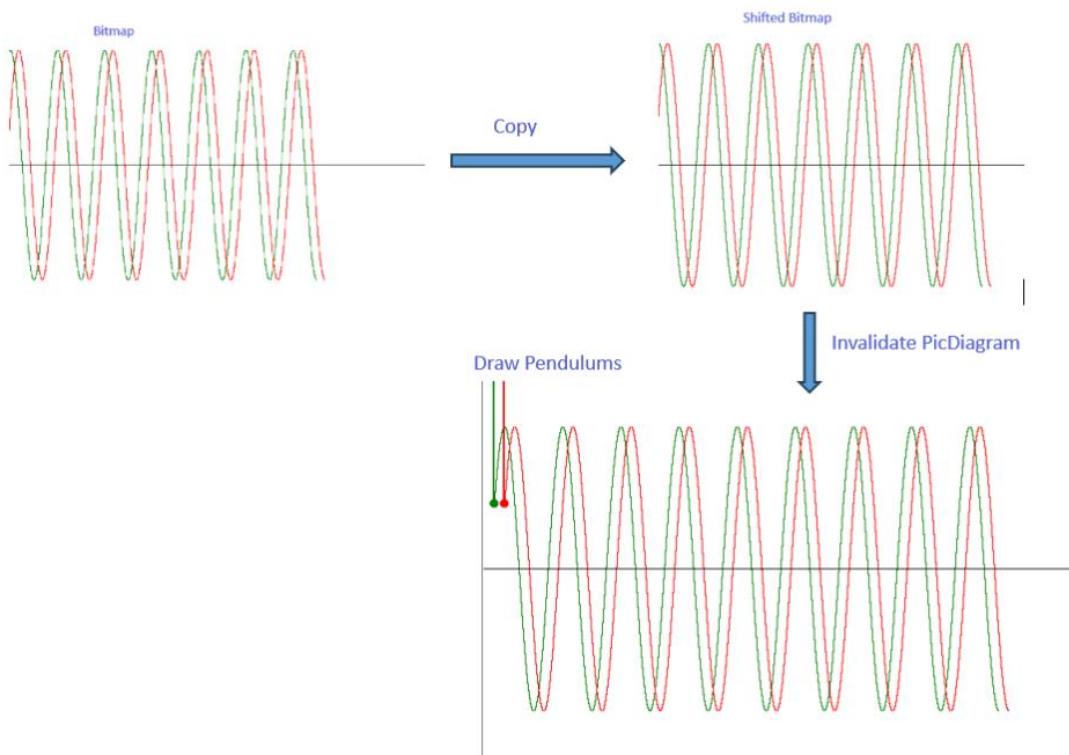
Die Picturebox *PicDiagram* zeichnet die aktuelle Position der Pendel auf. Das heisst, sie wird bei jedem Iterationsschritt geleert und das Pendel neu gezeichnet.

PicDiagram enthält eine Bitmap *MyBitmap* und darin wird die Position des Pendels permanent gezeichnet, was zur Aufzeichnung seiner Bahn führt.



`PicDiagram` und die dahinterliegende `MyBitmap`

Damit nun das «nach-rechts-schieben» funktioniert, geht man folgendermassen vor:

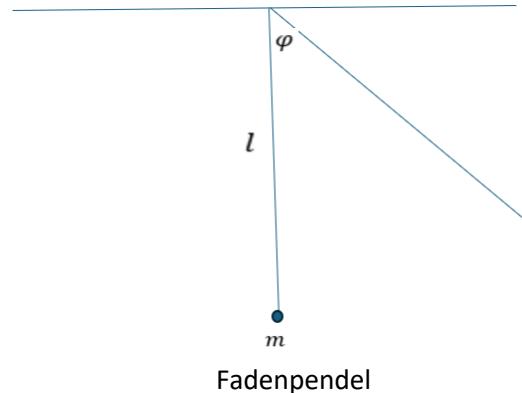


Die aktuelle Bitmap wird kopiert und dann geleert. Dann wird die Kopie der Bitmap wieder in die ursprüngliche Bitmap eingefügt, aber mit einem 1-Pixel Shift nach rechts. Durch `PicDiagram.Invalidate` die die geshiftete Bitmap in das PicDiagram übernommen. Anschliessend wird noch die aktuelle Position der Pendel in das PicDiagram eingezeichnet.

5.8. Übungsbeispiele

1. Beim expliziten Euler-Verfahren wurde die Amplitude zu wenig gedämpft, beim impliziten Euler-Verfahren zu stark. Wie ist es, wenn man beide Verfahren mittelt? Implementiere eine entsprechendes Verfahren im «Simulator» und unterscheide das Resultat. (Siehe Hinweise im letzten Kapitel «Implementierung eigener Varianten»).

2. Betrachte die Differentialgleichung $y' = y$ mit der Anfangsbedingung $y(0) = 1$. Bestimme explizit die Rekursionsformeln für das explizite und implizite Euler-Verfahren sowie für die Mittelpunktsregel.
3. Betrachte ein Fadenpendel mit Fadenlänge l und Pendelmasse m . Der Auslenkungswinkel sei $\varphi(t)$ wobei t die Zeit ist. g ist die Gravitationsbeschleunigung in Erdnähe $\approx 9.8 \text{ ms}^{-2}$.



Leite die Bewegungsgleichung her: $\ddot{\varphi} + \omega^2 \sin \varphi = 0$. Dabei ist $\omega = \sqrt{\frac{g}{l}}$.

Führe die Bewegungsgleichung durch die Transformation $u = \varphi, v = \dot{\varphi}$ über in ein System von zwei Differentialgleichungen erster Ordnung. Bestimme dann die Rekursionsformeln für die Mittelpunktsregel und auch für das Runge-Kutta Verfahren.

Implementiere beide Regeln im «Simulator» und vergleiche die Resultate.

6. Gekoppelte Pendel

6.1. Der Lagrange-Formalismus

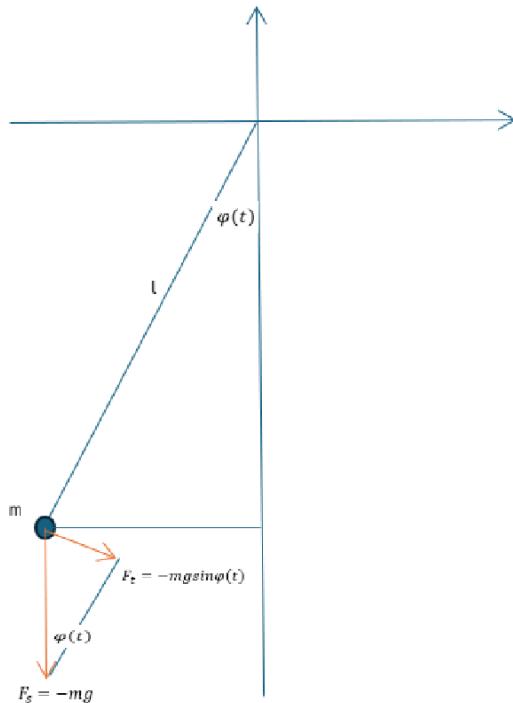
In den folgenden Abschnitten werden wir einige Beispiele von gekoppelten Pendeln betrachten und deren Dynamik im «Simulator» implementieren. Die Herleitung der entsprechenden Bewegungsgleichungen ist auf Basis der Newton'schen Kraftgesetze ziemlich schwierig bis unmöglich. Wir werden dafür den sogenannten Lagrange-Formalismus einsetzen. Dessen Herleitung aus dem Hamilton'schen Prinzip ist auf Basis der Mathematik am Gymnasium mit etwas Zusatzaufwand möglich, wenn man das Konzept der partiellen Ableitung und des totalen Differentials kennt und die Technik der partiellen Integration verwendet. Dabei kann es durchaus auch reizvoll sein, mit einfachen Beispielen in die Variationsrechnung einzutauchen. Eine gute Einführung findet man z.B. in [10].

Wir wollen hier aber auf eine Begründung des Lagrange-Formalismus verzichten und ihn nur soweit vorstellen, damit wir ihn in den folgenden Abschnitten verwenden können. Wir machen das anhand eines Beispiels.

Das Fadenpendel

Variante 1: Die Standardlösung

Der übliche Weg, die Bewegungsgleichung für das Fadenpendel herzuleiten, ist der Folgende:



Fadenpendel

Eine Masse m hängt an einem (masselosen) Faden der Länge l . Auf die Masse wirkt die Schwerkraft $F_s = -mg$ in negativer y -Richtung. g ist in diesem Kapitel immer die Gravitationsbeschleunigung in Erdnähe. Diese ist etwas unterschiedlich am Äquator und an den Polen, aber das spielt für unsere Untersuchungen keine Rolle. Wir rechnen immer mit $g = 9.8 \text{ ms}^{-2}$.

Zum Zeitpunkt t betrage die Auslenkung des Pendels $\varphi(t)$. Für die Bewegung des Masse ist nur die Tangentialkraft $F_t = -mgsin\varphi(t)$ relevant. Die Beschleunigung in Tangentialrichtung ist $l\ddot{\varphi}(t)$. Somit lautet das Newton'sche Bewegungsgesetz:

$$ml\ddot{\varphi} = -mgsin\varphi$$

Und man erhält als Bewegungsgleichung:

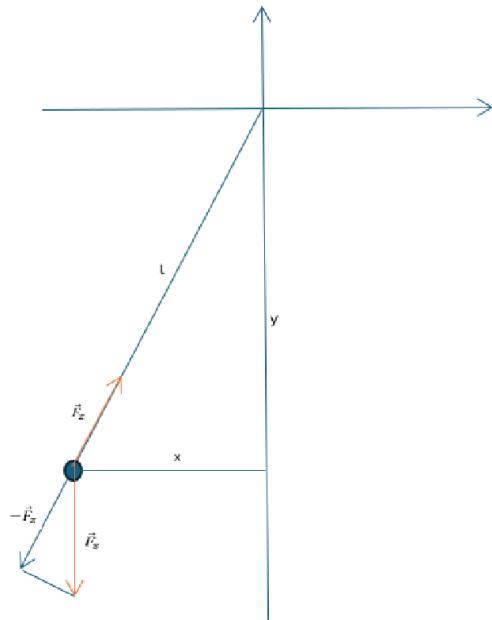
$$l\ddot{\varphi} + gsin\varphi = 0$$

Bei dieser ersten Variante hat man geeignete Koordinaten gewählt und die vom Faden ausgehende Zwangskraft dadurch eliminiert, weil man für die Beschleunigung der Masse nur die Tangentialkraft berücksichtigte.

Für eine explizite Lösung dieser Gleichung sind zusätzlich zwei Anfangsbedingungen nötig, zum Beispiel wenn das Pendel aus einer Ausgangslage mit dem Winkel φ_0 gehalten und dann losgelassen wird: $\varphi(0) = \varphi_0 > 0$, $\dot{\varphi}(0) = 0$

Variante 2: Explizite Berücksichtigung der vom Faden ausgehenden Zwangskraft

Es könnte ja sein, dass man nicht immer soviel Glück mit der Wahl des Koordinatensystems und der sich eliminierenden Zwangskraft hat. Wie sieht die Herleitung aus, wenn man letztere explizit berücksichtigt und ein kartesisches Koordinatensystem verwendet?



Fadenpendel in kartesischen Koordinaten

Der Ortsvektor des Massenpunktes zur Zeit t sei $\vec{r}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$. Die Schwerkraft, welche auf den Massenpunkt wirkt, ist $\vec{F}_g = \begin{pmatrix} 0 \\ -mg \end{pmatrix}$. Ferner gilt die Zwangsbedingung: $x^2 + y^2 = l^2$.

Für den Betrag der «Zwangskraft» \vec{F}_z , welche durch den Fadenausgeübt wird und dafür sorgt, dass das Pendel einen konstanten Abstand vom Aufhängepunkt hat, gilt wegen der Ähnlichkeit der beiden Dreiecke in der Skizze:

$$\frac{|\vec{F}_z|}{|\vec{F}_g|} = \frac{|y|}{l}$$

Deren Richtung ist dem Ortsvektor entgegengesetzt, zeigt also in Richtung des Einheitsvektors $\frac{-\vec{r}(t)}{l}$. Somit gilt für die Zwangskraft:

$$\vec{F}_z = mg \frac{-y}{l} \cdot \frac{1}{l} \begin{pmatrix} -x \\ -y \end{pmatrix} = \frac{mgy}{l^2} \begin{pmatrix} x \\ y \end{pmatrix}$$

Die Totalkraft, welche auf die Masse wirkt, ist dann:

$$\vec{F}_{total} = \begin{pmatrix} 0 \\ -mg \end{pmatrix} + \frac{mgy}{l^2} \begin{pmatrix} x \\ y \end{pmatrix} = \frac{mg}{l^2} \begin{pmatrix} xy \\ -l^2 + y^2 \end{pmatrix} = \frac{mg}{l^2} \begin{pmatrix} xy \\ -x^2 \end{pmatrix}$$

Somit lautet das Newton'sche Bewegungsgesetz:

$$\begin{cases} l^2 \ddot{x} = gxy \\ l^2 \ddot{y} = -gx^2 \end{cases}$$

Substituiert man y in der ersten Gleichung durch $y = -\sqrt{l^2 - x^2}$ (in unserer Skizze ist $y < 0$), erhält man das Differentialgleichungssystem:

$$(*) \begin{cases} l^2 \ddot{x} = -gx\sqrt{l^2 - x^2} \\ l^2 \ddot{y} = -gx^2 \end{cases}$$

Um hier weiterzukommen, liegt der Ansatz $x(t) = l \cdot \sin\varphi(t)$ auf der Hand. Wegen der Zwangsbedingung ist dann $y(t) = -l \cdot \cos\varphi(t)$, wobei in beiden Fällen $\varphi(t)$ eine noch zu bestimmende Funktion ist.

Um x und y im Gleichungssystem zu eliminieren, berechnen wir auch die Ableitungen:

$$\begin{aligned} \dot{x} &= l\dot{\varphi}\cos\varphi, \ddot{x} = l\ddot{\varphi}\cos\varphi - l\dot{\varphi}^2\sin\varphi \\ \dot{y} &= l\dot{\varphi}\sin\varphi, \ddot{y} = l\ddot{\varphi}\sin\varphi + l\dot{\varphi}^2\cos\varphi \end{aligned}$$

Wir setzen alles in $(*)$ ein und erhalten:

$$\begin{cases} l^3 \ddot{\varphi}\cos\varphi - l^3 \dot{\varphi}^2\sin\varphi = -gl\sin\varphi \cdot l\cos\varphi \\ l^3 \ddot{\varphi}\sin\varphi + l^3 \dot{\varphi}^2\cos\varphi = -gl^2\sin^2\varphi \end{cases}$$

Nun bringen wir alles auf die linke Seite und dividieren die erste Gleichung durch $l^2\sin\varphi$. Die zweite Gleichung dividieren wir durch $l^2\cos\varphi$ (ohne dass wir jetzt die Fälle, an denen diese Ausdrücke Null sind, speziell behandeln). Dann erhalten wir die Gleichungen:

$$\begin{cases} l\ddot{\varphi} \frac{\cos\varphi}{\sin\varphi} - l\dot{\varphi}^2 + g\cos\varphi = 0 \\ l\ddot{\varphi} \frac{\sin\varphi}{\cos\varphi} + l\dot{\varphi}^2 + g \frac{\sin^2\varphi}{\cos\varphi} = 0 \end{cases}$$

Addiert man beide Gleichungen, ergibt das:

$$l\ddot{\varphi} \left(\frac{\cos\varphi}{\sin\varphi} + \frac{\sin\varphi}{\cos\varphi} \right) + g \left(\cos\varphi + \frac{\sin^2\varphi}{\cos\varphi} \right) = 0$$

Nun multiplizieren wir diese Gleichung mit dem Faktor $\sin\varphi \cdot \cos\varphi$ und erhalten:

$$l\ddot{\varphi}(\cos^2\varphi + \sin^2\varphi) + g \cdot \sin\varphi(\cos^2\varphi + \sin^2\varphi) = 0$$

Das liefert wieder die bekannte Gleichung:

$$l\ddot{\varphi} + g \sin \varphi = 0$$

Allerdings war der Aufwand sehr viel grösser.

Einschub: Der Lagrange Formalismus

Der Lagrange-Formalismus erlaubt es, mit Parametern zu arbeiten, welche ein System eindeutig beschreiben, aber sonst beliebig gewählt werden können. Man spricht dann von *generalisierten Koordinaten*. Im Idealfall (und das wird in den folgenden Abschnitten so sein) kann man diese so wählen, dass man keine Zwangsbedingungen zu berücksichtigen hat, sondern dass diese bereits implizit mit der Wahl der generalisierten Koordinaten berücksichtigt sind. Dann entspricht die Anzahl dieser Koordinaten der Anzahl der unabhängigen Freiheitsgrade eines Systems.

Die generalisierten Koordinaten sind im allgemeinen zeitabhängig und man bezeichnet sie mit $q_1(t), q_2(t), q_3(t), \dots$. Die kartesischen Koordinaten (von vielleicht mehreren Massenpunkten im System) hängen dann von den generalisierten Koordinaten ab. Für die i-te kartesische Koordinate gilt dann:

$$x_i(t) = x_i(q_1(t), q_2(t), \dots) = x_i(\vec{q})$$

Für uns wird es genügen, nur Systeme zu betrachten, bei denen sich die äusseren (das heisst nicht-Zwangs-) Kräfte aus einem Potential $V(\vec{q})$ ableiten lassen, in denen also der Energiesatz gilt.

Nun definiert man die Lagrange-Funktion als Differenz von kinetischer Energie T und potenzieller Energie V . Sie hängt im allgemeinen ab von $\vec{q}, \dot{\vec{q}}$ und t .

$$L(\vec{q}, \dot{\vec{q}}, t) = T - V$$

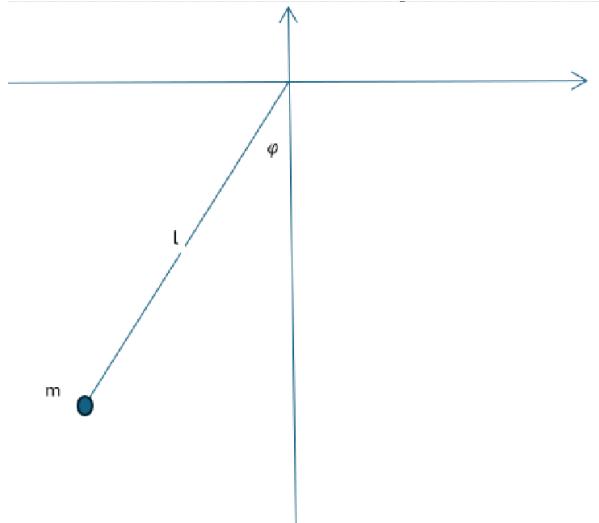
Da ein Potential V nur bis auf eine Konstante bestimmt ist, gilt dies auch für die Lagrange-Funktion.

Ein Massenpunkt wird sich nun auf einer Bahn bewegen, so dass das sogenannte Hamilton'sche Prinzip der stationären Wirkung minimal wird. Da wir auf eine Herleitung der Lagrange-Gleichungen verzichten, gehen wir darauf nicht näher ein. Das Resultat ist, dass komponentenweise folgende Gleichungen gelten:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} = \frac{\partial L}{\partial q_i}$$

Und zwar für alle $1 \leq i \leq \text{Anzahl generalisierte Koordinaten}$.

Variante 3: Das Fadenpendel und der Lagrange-Formalismus



Fadenpendel mit generalisierter Koordinate φ

Die kartesischen Koordinaten hängen dann wie folgt von φ ab:

$$\begin{pmatrix} x(\varphi(t)) \\ y(\varphi(t)) \end{pmatrix} = l \begin{pmatrix} \sin\varphi(t) \\ -\cos\varphi(t) \end{pmatrix}$$

Das Potential ist das Gravitationsfeld in Erdnähe und gegeben durch $V(\varphi) = mgl(1 - \cos\varphi)$ wenn wir es normieren mit $V(0) = 0$.

Somit lautet die Lagrange-Funktion (der erste Term ist die kinetische Energie):

$$L(\varphi, \dot{\varphi}, t) = \frac{1}{2} m(\dot{x}^2 + \dot{y}^2) - mgl(1 - \cos\varphi) = \frac{m}{2} l^2 \dot{\varphi}^2 - mgl(1 - \cos\varphi)$$

Die linke Seite der Lagrange-Gleichung ist dann:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\varphi}} = \frac{d}{dt} ml^2 \dot{\varphi} = ml^2 \ddot{\varphi}$$

Die rechte Seite der Lagrange-Gleichung ist:

$$\frac{\partial L}{\partial \varphi} = -mglsin\varphi$$

Wenn wir beide Seiten gleichsetzen, erhalten wir:

$$ml^2 \ddot{\varphi} = -mglsin\varphi$$

Und damit wieder die bekannte Gleichung:

$$l\ddot{\varphi} + gsin\varphi = 0$$

□

Wir wollen als weiteres Beispiel den Fall eines Massepunktes betrachten, auf den keine Kraft wirkt. Dann ist V konstant und bei entsprechender Normierung $V \equiv 0$.

Da auch keine Zwangskräfte wirken, sind gerade kartesische Koordinaten naheliegende Koordinaten, um die Bewegung des Massepunktes zu beschreiben. Die Lagrange-Funktion besteht dann nur aus der kinetischen Energie:

$$L = \frac{1}{2} m(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)$$

Dann gilt:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{x}} = \frac{d}{dt} m\dot{x} = m\ddot{x} = \frac{\partial L}{\partial x} = 0$$

Es ist also $m\dot{x} = \text{konstant}$. Aber das ist nichts anderes als die Impulskomponente in x-Richtung.

Dasselbe gilt für die Koordinaten y,z.

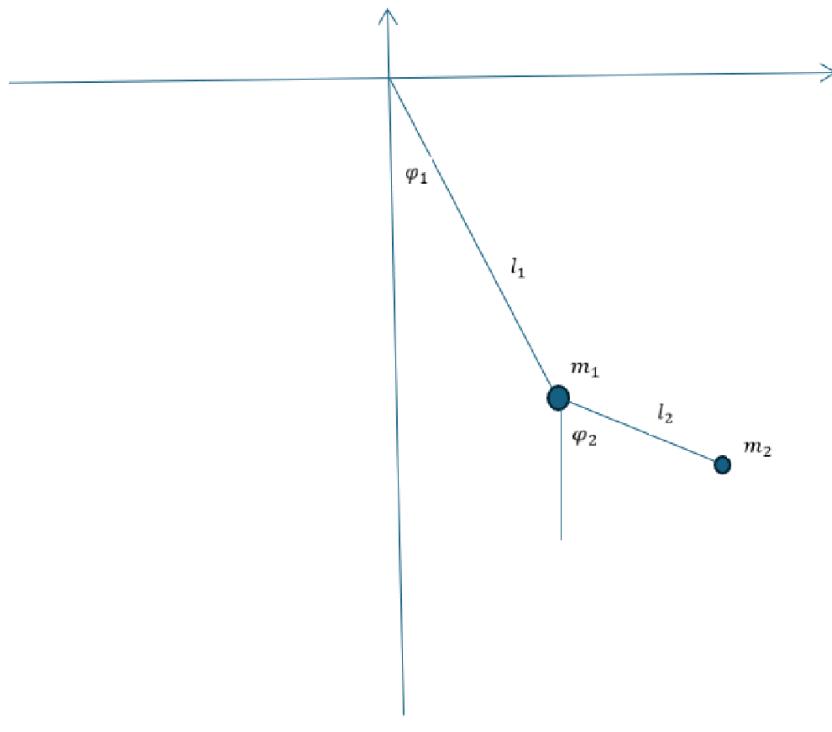
Folgerung: Im Kräftefreien Raum ist der Impuls eines Massepunktes konstant.

□

Hier stellt sich die Frage, ob die Lagrange-Gleichung von der Wahl der generalisierten Koordinaten abhängt. Das ist nicht der Fall und das hat damit zu tun, dass das Hamilton'sche Prinzip der stationären Wirkung unabhängig von dieser Wahl ist. Es ist ähnlich wie bei der Bestimmung eines Extremalwertes einer Funktion. Die Frage, wie man aus möglichst wenig Blech eine zylindrische Büchse mit vorgegebenem Volumen konstruiert, hängt auch nicht von der Wahl der Koordinaten ab. Ein Beweis der Unabhängigkeit des Hamilton-Prinzips von der Koordinatenwahl würde aber hier zu weit führen. Man findet eine Begründung dafür in [10].

6.2. Das Doppelpendel

Als erstes Beispiel eines gekoppelten Pendels betrachten wir im «Simulator» das Doppelpendel.



Doppelpendel

Die generalisierten Koordinaten sind $\begin{pmatrix} \varphi_1(t) \\ \varphi_2(t) \end{pmatrix}$. Sie beschreiben das System zum Zeitpunkt t vollständig. $l_1 > 0, l_2 > 0$ bleiben während der Bewegung konstant.

Nun werden wir um einen gewissen Rechenaufwand nicht herumkommen.

Der Ortsvektor der Masse m_1 ist:

$$\vec{r}_1 = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = l_1 \begin{pmatrix} \sin\varphi_1 \\ -\cos\varphi_1 \end{pmatrix}, \dot{\vec{r}}_1 = l_1 \dot{\varphi}_1 \begin{pmatrix} \cos\varphi_1 \\ \sin\varphi_1 \end{pmatrix}$$

Und der Ortsvektor der Masse m_2 :

$$\begin{aligned} \vec{r}_2 &= \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = l_1 \begin{pmatrix} \sin\varphi_1 \\ -\cos\varphi_1 \end{pmatrix} + l_2 \begin{pmatrix} \sin\varphi_2 \\ -\cos\varphi_2 \end{pmatrix} \\ \dot{\vec{r}}_2 &= l_1 \dot{\varphi}_1 \begin{pmatrix} \cos\varphi_1 \\ \sin\varphi_1 \end{pmatrix} + l_2 \dot{\varphi}_2 \begin{pmatrix} \cos\varphi_2 \\ \sin\varphi_2 \end{pmatrix} \end{aligned}$$

Die kinetische Energie ist gegeben durch:

$$\begin{aligned} E_{kin} &= \frac{1}{2} m_1 |\dot{\vec{r}}_1|^2 + \frac{1}{2} m_2 |\dot{\vec{r}}_2|^2 \\ &= \frac{1}{2} m_1 l_1^2 \dot{\varphi}_1^2 + \frac{1}{2} m_2 \{l_1^2 \dot{\varphi}_1^2 + 2l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 (\cos\varphi_1 \cos\varphi_2 + \sin\varphi_1 \sin\varphi_2) + l_2^2 \dot{\varphi}_2^2\} \\ &= \frac{1}{2} m_1 l_1^2 \dot{\varphi}_1^2 + \frac{1}{2} m_2 \{l_1^2 \dot{\varphi}_1^2 + 2l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \cos(\varphi_1 - \varphi_2) + l_2^2 \dot{\varphi}_2^2\} \end{aligned}$$

Und die potenzielle Energie:

$$V = m_1 g y_1 + m_2 g y_2 = -m_1 g l_1 \cos\varphi_1 - m_2 g l_1 \cos\varphi_1 - m_2 g l_2 \cos\varphi_2$$

Damit hat man als Lagrange-Funktion:

$$\begin{aligned} L &= E_{kin} - V \\ &= \frac{1}{2} (m_1 + m_2) l_1^2 \dot{\varphi}_1^2 + m_2 l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \cos(\varphi_1 - \varphi_2) + \frac{1}{2} m_2 l_2^2 \dot{\varphi}_2^2 \\ &\quad + g(m_1 + m_2) l_1 \cos\varphi_1 + g m_2 l_2 \cos\varphi_2 \end{aligned}$$

Die Lagrange-Gleichung für die erste generalisierte Koordinate lautet:

$$\begin{aligned} \frac{d}{dt} \frac{\partial L}{\partial \dot{\varphi}_1} &= \frac{\partial L}{\partial \varphi_1} \\ \frac{d}{dt} \frac{\partial L}{\partial \dot{\varphi}_1} &= \frac{d}{dt} \{l_1^2 (m_1 + m_2) \dot{\varphi}_1 + m_2 l_1 l_2 \cos(\varphi_1 - \varphi_2) \dot{\varphi}_2\} \\ &= l_1^2 (m_1 + m_2) \ddot{\varphi}_1 + m_2 l_1 l_2 \cos(\varphi_1 - \varphi_2) \ddot{\varphi}_2 - m_2 l_1 l_2 \sin(\varphi_1 - \varphi_2) \dot{\varphi}_2 (\dot{\varphi}_1 - \dot{\varphi}_2) \\ \frac{\partial L}{\partial \varphi_1} &= -m_2 l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \sin(\varphi_1 - \varphi_2) - g(m_1 + m_2) l_1 \sin\varphi_1 \end{aligned}$$

Wir setzen diese Ausdrücke in die Lagrange-Gleichung ein und bringen alles nach links. Ferner kann durch l_1 gekürzt werden. Das liefert:

$$\begin{aligned} l_1 (m_1 + m_2) \ddot{\varphi}_1 + m_2 l_2 \cos(\varphi_1 - \varphi_2) \ddot{\varphi}_2 \\ - m_2 l_2 \sin(\varphi_1 - \varphi_2) \dot{\varphi}_2 (\dot{\varphi}_1 - \dot{\varphi}_2) + m_2 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \sin(\varphi_1 - \varphi_2) + g(m_1 + m_2) \sin\varphi_1 = 0 \end{aligned}$$

Multipliziert man den dritten Summanden aus, hebt sich ein Teil mit am vierten Summanden auf und es bleibt:

$$l_1 (m_1 + m_2) \ddot{\varphi}_1 + m_2 l_2 \{\cos(\varphi_1 - \varphi_2) \ddot{\varphi}_2 + \sin(\varphi_1 - \varphi_2) \dot{\varphi}_2^2\} + g(m_1 + m_2) \sin\varphi_1 = 0$$

Wir setzen:

$$\Delta\varphi := \varphi_1 - \varphi_2, \mu := \frac{m_2}{m_1 + m_2}$$

Wenn $m_1 > 0$ ist $\mu < 1$.

Damit erhalten wir:

$$(1) \quad l_1\ddot{\varphi}_1 + \mu l_2 \cos\Delta\varphi \ddot{\varphi}_2 + \mu l_2 \sin\Delta\varphi \dot{\varphi}_2^2 + g \sin\varphi_1 = 0$$

Die Lagrange-Gleichung für die zweite generalisierte Koordinate lautet analog:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\varphi}_2} = \frac{\partial L}{\partial \varphi_2}$$

Eine entsprechende Rechnung, welche wir gerne dem Leser als Übung überlassen, liefert die Gleichung:

$$(2) \quad l_2\ddot{\varphi}_2 + l_1 \cos\Delta\varphi \ddot{\varphi}_1 - l_1 \sin\Delta\varphi \dot{\varphi}_1^2 + g \sin\varphi_2 = 0$$

Beide Gleichungen haben eine hohe Symmetrie: Vertauscht man die Indices $1 \leftrightarrow 2$, dann geht die eine fast in die andere über. Sie unterscheiden sich dann nur um den Faktor μ und $\Delta\varphi$ geht über in $-\Delta\varphi$.

Wenn wir die zweite Gleichung (2) mit $\mu \cos\Delta\varphi$ multiplizieren, erhalten wir die Gleichungen:

$$\begin{cases} l_1\ddot{\varphi}_1 + \mu l_2 \cos\Delta\varphi \ddot{\varphi}_2 + \mu l_2 \sin\Delta\varphi \dot{\varphi}_2^2 + g \sin\varphi_1 = 0 \\ l_1 \mu \cos^2 \Delta\varphi \ddot{\varphi}_1 + \mu l_2 \cos\Delta\varphi \ddot{\varphi}_2 - \mu l_1 \sin\Delta\varphi \cos\Delta\varphi \dot{\varphi}_1^2 + \mu \cos\Delta\varphi g \sin\varphi_2 = 0 \end{cases}$$

Nun subtrahieren wir die untere Gleichung von der oberen und erhalten, wenn wir nach $\ddot{\varphi}_1$ auflösen:

$$\ddot{\varphi}_1 = \frac{-\mu \sin\Delta\varphi (l_2\ddot{\varphi}_2^2 + l_1 \cos\Delta\varphi \dot{\varphi}_1^2) + g(\mu \cos\Delta\varphi \sin\varphi_2 - \sin\varphi_1)}{l_1(1 - \mu \cos^2 \Delta\varphi)}$$

Wenn wir die erste Gleichung (1) mit $\cos\Delta\varphi$ multiplizieren, erhalten wir die Gleichungen:

$$\begin{cases} l_1 \cos\Delta\varphi \ddot{\varphi}_1 + \mu l_2 \cos^2 \Delta\varphi \ddot{\varphi}_2 + \mu l_2 \sin\Delta\varphi \cos\Delta\varphi \dot{\varphi}_2^2 + g \cos\Delta\varphi \sin\varphi_1 = 0 \\ l_1 \cos\Delta\varphi \ddot{\varphi}_1 + l_2 \ddot{\varphi}_2 - l_1 \sin\Delta\varphi \dot{\varphi}_1^2 + g \sin\varphi_2 = 0 \end{cases}$$

Wieder subtrahieren wir die untere Gleichung von der oberen:

$$l_2(\mu \cos^2 \Delta\varphi - 1)\ddot{\varphi}_2 + \mu l_2 \sin\Delta\varphi \cos\Delta\varphi \dot{\varphi}_2^2 + g \cos\Delta\varphi \sin\varphi_1 + l_1 \sin\Delta\varphi \dot{\varphi}_1^2 - g \sin\varphi_2 = 0$$

Aufgelöst nach $\ddot{\varphi}_2$ liefert das:

$$\ddot{\varphi}_2 = \frac{\sin\Delta\varphi(l_1\dot{\varphi}_1^2 + \mu l_2 \cos\Delta\varphi \dot{\varphi}_2^2) + g(\cos\Delta\varphi \sin\varphi_1 - \sin\varphi_2)}{l_2(1 - \mu \cos^2 \Delta\varphi)}$$

Wenn $m_1 > 0$ ist $\mu < 1$ und damit der Nenner in beiden Fällen grösser als 0.

Die Schwingungsfrequenz eines isolierten Pendels wäre $\omega_i = \sqrt{\frac{g}{l_i}}$, $i \in \{1,2\}$.

Um die Gleichungen für $\ddot{\varphi}_1$ und $\ddot{\varphi}_2$ in ein Gleichungssystem erster Ordnung zu überführen, setzen wir:

$$u_1 = \varphi_1, v_1 = \dot{\varphi}_1, u_2 = \varphi_2, v_2 = \dot{\varphi}_2$$

$$\begin{cases} \dot{u}_1 = v_1 =: f_1(t, u_1, v_1, u_2, v_2) \\ \dot{v}_1 = \frac{-\mu \sin \Delta u (l_2 v_2^2 + l_1 \cos \Delta u v_1^2) + g(\mu \cos \Delta u \sin u_2 - \sin u_1)}{l_1(1 - \mu \cos^2 \Delta u)} =: g_1(t, u_1, v_1, u_2, v_2) \\ \dot{u}_2 = v_2 =: f_2(t, u_1, v_1, u_2, v_2) \\ \dot{v}_2 = \frac{\sin \Delta u (l_1 v_1^2 + \mu l_2 \cos \Delta u v_2^2) + g(\cos \Delta u \sin u_1 - \sin u_2)}{l_2(1 - \mu \cos^2 \Delta u)} =: g_2(t, u_1, v_1, u_2, v_2) \end{cases}$$

Dabei ist $\Delta u = u_1 - u_2$.

Das vierstufige Runge-Kutta Verfahren bei einer Schrittweite h ist für den n+1-ten Schritt etwas aufwendig, weil wir vier Parameter u_1, v_1, u_2, v_2 berücksichtigen müssen. Damit die Implementierung im «Simulator» dokumentiert ist, bereiten wir diese in folgendem Algorithmus vor. Darin sind $u_{1n}, v_{1n}, u_{2n}, v_{2n}$ die Werte der Parameter nach dem n-ten Iterationsschritt. Die (konstante) Schrittweite ist d, welche genügend klein gewählt werden muss. Nun versuchen wir, die Darstellung etwas kompakt zu halten. Nacheinander führen wir folgende Schritte durch:

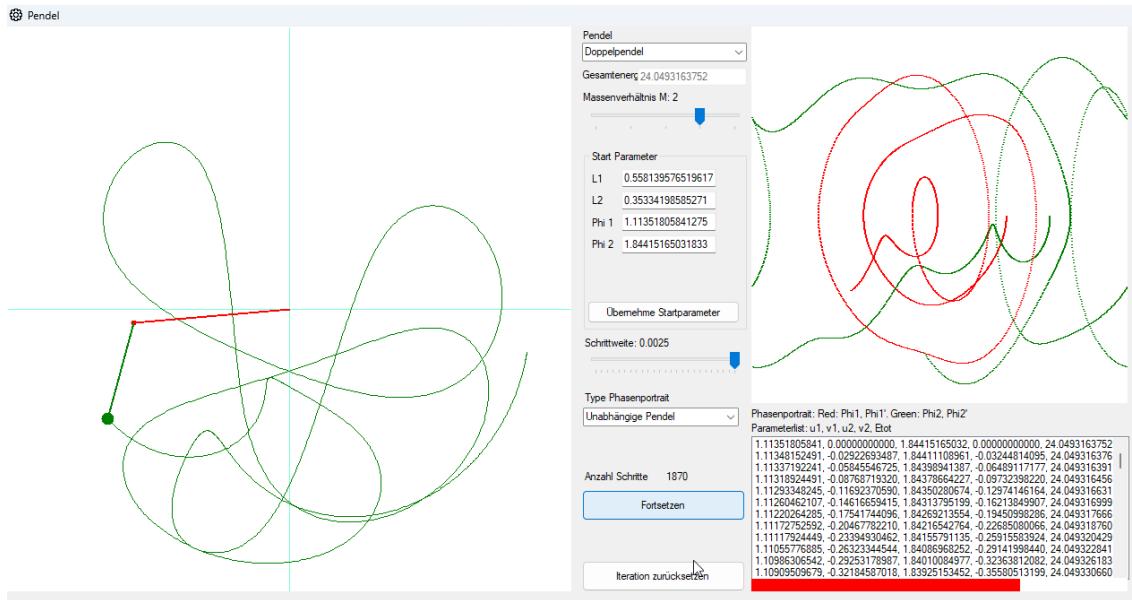
$$\begin{aligned} \vec{x}_{1n} &:= (u_{1n}, v_{1n}, u_{2n}, v_{2n}) \\ \begin{cases} k_{i1} := f_i(t_n, \vec{x}_{1n}) \\ h_{i1} := g_i(t_n, \vec{x}_{1n}) \end{cases}, i \in \{1,2\} \\ \vec{x}_{2n} &:= (u_{1n} + \frac{d}{2} k_{11}, v_{1n} + \frac{d}{2} h_{11}, u_{2n} + \frac{d}{2} k_{21}, v_{2n} + \frac{d}{2} h_{21}) \\ \begin{cases} k_{i2} := f_i(t_n + \frac{d}{2}, \vec{x}_{2n}) \\ h_{i2} := g_i(t_n + \frac{d}{2}, \vec{x}_{2n}) \end{cases}, i \in \{1,2\} \\ \vec{x}_{3n} &:= (u_{1n} + \frac{d}{2} k_{12}, v_{1n} + \frac{d}{2} h_{12}, u_{2n} + \frac{d}{2} k_{22}, v_{2n} + \frac{d}{2} h_{22}) \\ \begin{cases} k_{i3} := f_i(t_n + \frac{d}{2}, \vec{x}_{3n}) \\ h_{i3} := g_i(t_n + \frac{d}{2}, \vec{x}_{3n}) \end{cases}, i \in \{1,2\} \\ \vec{x}_{4n} &:= (u_{1n} + k_{13}, v_{1n} + h_{13}, u_{2n} + k_{23}, v_{2n} + h_{23}) \\ \begin{cases} k_{i4} := f_i(t_n + d, \vec{x}_{4n}) \\ h_{i4} := g_i(t_n + d, \vec{x}_{4n}) \end{cases}, i \in \{1,2\} \\ \begin{cases} u_{i(n+1)} = u_{in} + \frac{t_{n+1} - t_n}{6} (k_{i1} + 2k_{i2} + 2k_{i3} + k_{i4}) \\ v_{i(n+1)} = v_{in} + \frac{d(h_{i1} + 2h_{i2} + 2h_{i3} + h_{i4})}{6} \end{cases}, i \in \{1,2\} \end{aligned}$$

Wir werden beim Experimentieren sehen, dass das Runge Kutta-Verfahren für das Doppelpendel nur bedingt brauchbar ist, nämlich nur für niedrige Energiewerte bzw. Pendelausschläge.

6.3. Implementierung des Doppelpendels

Die *FrmPendulum* bietet wie bereits ähnliche Formen die Darstellung von aktuellen Pendelpositionen in *PicPendulum* und die Aufzeichnung der Bahn der Pendel in der Bitmap *MapPendulum*. Auf der

rechten Seite wird die Bewegung der Pendel in *PicPhasePortrait* als Phasendiagramm aufgezeichnet sowie die einzelnen Parameterwerte in *LstParameterList* protokolliert.



FrmPendulum

Dann können bis zu 6 Parameter angegeben werden, welche je nach Pendel konstant oder variabel sind (das bestimmt die Implementierung des Pendels). Diese Parameter werden automatisch gesetzt, wenn die Startposition der Pendel manuell mit der Maus gesetzt wird. Sie können aber auch manuell eingegeben werden und den Pendeln übergeben werden. Der Faktor C wird je nach Pendel definiert und ebenso ein optionaler zusätzlicher Parameter (*AdditionalParameter*), der durch ein Schieberegister gesetzt wird.

Die Kommunikation zwischen *FrmPendulum* und der Implementierung der einzelnen Pendel ist durch das Interface *IPendulum* definiert.

Zum besseren Verständnis des Codes hier einige Erklärungen:

- Die frei wählbaren Parameter sind in der *FrmPendulum* mit *TxtP1...TxtP6* bezeichnet.
- Via Interface werden diese an die Pendelklasse (z.B. *ClsDoublePendulum*) übergeben als Vektoren *MyConstants* und *MyVariables*
- Eine oder mehrere Routinen *SetandDrawStartPosition* unterstützen das Festlegen der Startpositionen mit der Maus. Sie setzen dann auch *MyConstants* und *MyVariables*, deren Werte in die *FrmPendulum* mit *TxtP1...TxtP6* eingetragen werden.
- Die Positionen der Pendel werden als *ClsMathPoint* in *Position* gehalten und als Startwert ebenfalls durch *SetandDrawStartPosition* gesetzt.
- Die Position der Pendel wird in *DrawPendulum* laufend auf Grund der *Position* gezeichnet.

Zur Iteration folgende Bemerkungen:

- Die relevanten Iterationsparameter sind in Anlehnung an die Formeln des Runge Kutta-Verfahrens zum Beispiel beim Doppelpendel u_1, v_1, u_2, v_2 . Deren Startwert wird auch durch *SetandDrawStartPosition* gesetzt.
- Am Beginn eines Iterationsschrittes wird *OldPosition = Position* gesetzt. *OldPosition* ist also die alte Position des Pendels vor dem Iterationsschritt.

- Für die Iteration werden beim Doppelpendel in Anlehnung an die mathematischen Formeln ein *ClVector* $x(3)$ gebraucht. Er spielt die Rolle der einzelnen \vec{x}_{in} .
- $k_{11}, k_{12}, k_{13}, k_{14}$ wird als *ClVector(3)* geführt. Ebenso k_{21}, k_{22} und k_{23} .
- Am Ende jedes Iterationsschrittes wird die aktuelle Position des Pendels auf Grund von *Position* gezeichnet und die Pendelbahn in *DrawTrack* auf Grund von *OldPosition* und *Position*.

Testen und Überwachung

Falls die Checkbox «Testmodus» aktiviert ist, dann werden die Funktionen im Runge Kutta Verfahren so ersetzt, dass beide Pendel unabhängig voneinander schwingen wie normale Fadenpendel (aber mit möglicherweise grosser Auslenkung). Für beide Pendel wird hier die Differentialgleichung

$$\ddot{\varphi}_i = \frac{-g \sin \varphi_i}{l_i}, i \in \{1,2\}$$

verwendet.

Es ist offensichtlich, dass das Runge Kutta-Verfahren im Falle des Doppelpendels unzuverlässig ist. Bei kleinen Schwingungen scheint es den Erwartungen an ein Doppelpendel zu entsprechen. Bei grossen Schwingungen erscheinen manche Bewegungen unnatürlich. Zu beachten ist auch, dass die Simulation nicht den tatsächlichen zeitlichen Verlauf widerspiegelt, sondern von der Schnelligkeit jedes Berechnungsschrittes beim Runge Kutta-Verfahren abhängt.

Was man aber als Minimum verlangen kann, ist, dass die Gesamtenergie des Doppelpendels während der Bewegung konstant bleibt, mindestens einigermassen. Das liefert zwar keine Garantie für die Zuverlässigkeit des Verfahrens. Aber umgekehrt ist ersichtlich, wenn das Verfahren «aus dem Ruder» läuft.

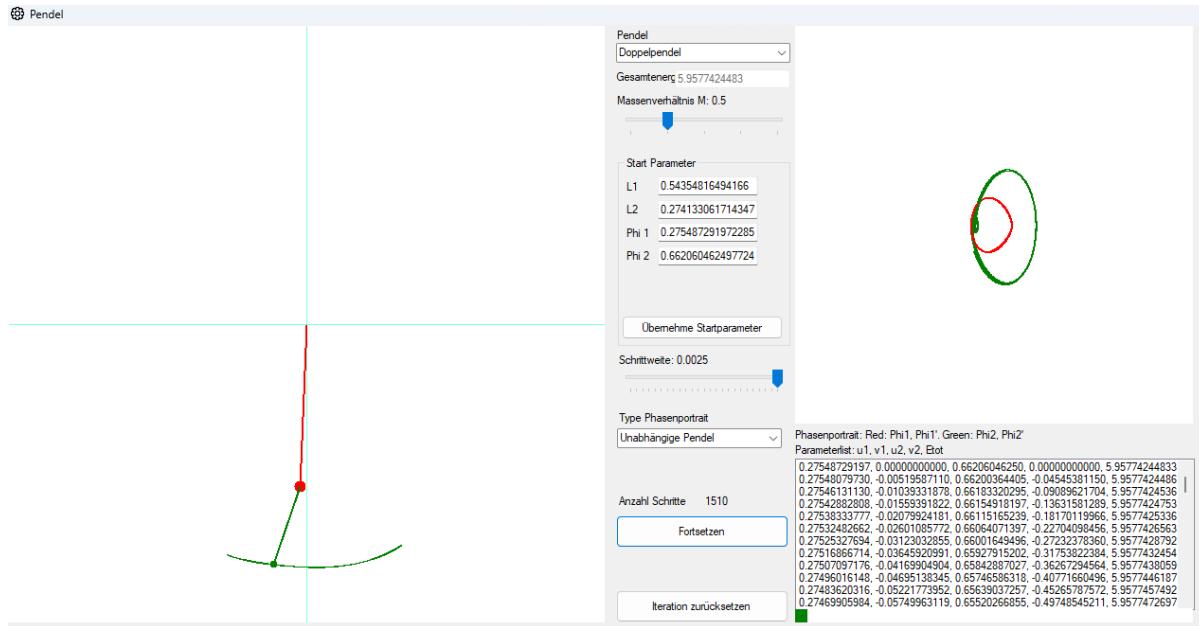
Für die kinetische Energie des Doppelpendels gilt:

$$E_{kin} = \frac{1}{2} m_1 l_1^2 \dot{\varphi}_1^2 + \frac{1}{2} m_2 \{ l_1^2 \dot{\varphi}_1^2 + 2l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \cos(\varphi_1 - \varphi_2) + l_2^2 \dot{\varphi}_2^2 \}$$

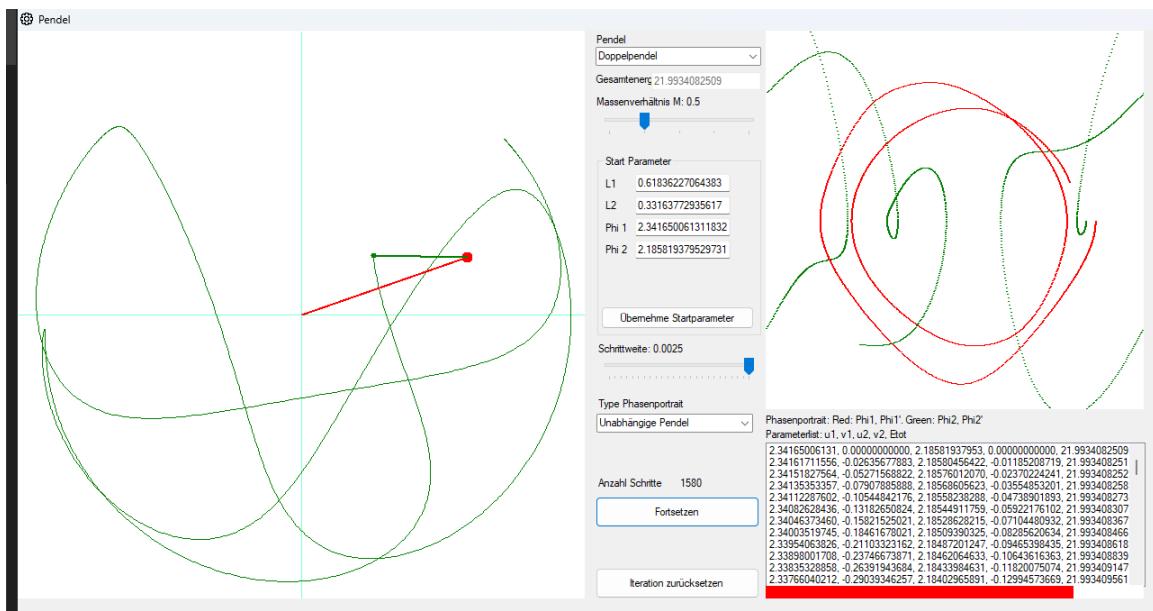
Für die potenzielle Energie setzen wir das Null-Niveau auf $y = -1$. Damit wird:

$$E_{pot} = g \{ (1 - l_1 \cos \varphi_1)(m_1 + m_2) - m_2 l_2 \cos \varphi_2 \}$$

Während der Iteration wird die aktuelle Energie des Doppelpendels laufend berechnet und mit der Startenergie verglichen. Das Resultat wird im obigen Bild durch einen waagrechten Balken rechts unten dargestellt. Das Gesamtintervall entspricht dem Bereich der möglichen Startenergie und der farbige Balken der momentanen Energie des Doppelpendels. Wenn dieser Balken grün ist, bedeutet dies, dass die Abweichung der Pendelenergie kleiner als 10% relativ zum Gesamtintervall ist. Ist die Pendelenergie höher, wird der Balken rot. Ist sie tiefer, wird der Balken violett.



Testmodus und Überwachung der Gesamtenergie rechts bei tiefer Energie:
Die Gesamtenergie bleibt im grünen Bereich



Testmodus und Überwachung der Gesamtenergie rechts bei hoher Energie:
Die Gesamtenergie ist manchmal zu hoch (rot) oder zu tief (violet)

Bei der Implementierung setzen wir $m_1 = 1$ und $m_2 = M$. Ferner setzen wir:

$$\varphi_i = u_i, \dot{\varphi}_i = v_i, i \in \{1, 2\}$$

Damit erhalten wir für die Energie:

$$E_{tot} = \frac{1}{2} l_1^2 \dot{\varphi}_1^2 + \frac{1}{2} M \{ l_1^2 \dot{\varphi}_1^2 + 2l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \cos(\varphi_1 - \varphi_2) + l_2^2 \dot{\varphi}_2^2 \} + g \{ (1 - l_1 \cos \varphi_1)(1 + M) - M l_2 \cos \varphi_2 \}$$

Der Start des Doppelpendels erfolgt aus der Ruhelage. Dort ist $E_{kin} = 0$. Die potenzielle Energie ist minimal für $\varphi_1 = \varphi_0 = 0$ und maximal für $\varphi_1 = \varphi_0 = \pi$. Wir bezeichnen diese Werte mit:

$$E_{min} = g\{(1 - l_1)(1 + M) - Ml_2\}$$

$$E_{max} = g\{(1 + l_1)(1 + M) + Ml_2\}$$

Die Pendelenergie liegt also im Intervall $[E_{min}, E_{max}]$. Die aktuelle Pendelenergie überschreitet die Limite von 10%, wenn:

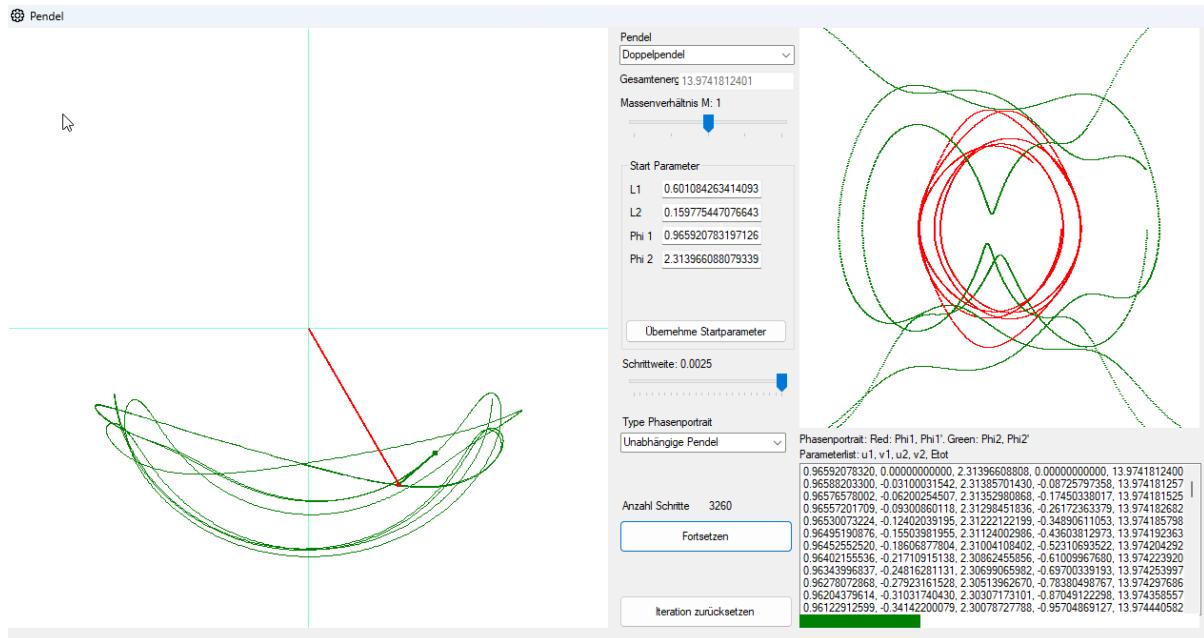
$$E_{aktuell} \geq E_{Start} + 0.1(E_{max} - E_{min})$$

Und unterschreitet sie, falls:

$$E_{aktuell} \leq E_{Start} - 0.1(E_{max} - E_{min})$$

6.4. Untersuchung des Phasenraumes für das Doppelpendel

Wir haben im früheren Kapitel die Bewegung der Pendel einzeln im Phasenraum dargestellt. Das heisst, ausgehend von einem Startpunkt haben wir die Kurven $(\varphi_1, \dot{\varphi}_1)$ und $(\varphi_2, \dot{\varphi}_2)$ einzeln aufgezeichnet. Das ergibt ein Bild der Bewegung mit einer konstanten Energie ausgehend vom Startpunkt. Wenn man nun eine Übersicht über das Verhalten des Doppelpendels quer durch alle möglichen Energiezustände gewinnen will, ist diese Darstellung ungeeignet.



Darstellung der einzelnen Pendel im Phasenraum

Eine bessere Möglichkeit, das Verhalten eines dynamischen Systems im Phasenraum zu untersuchen, ist der sogenannte Poincaré-Schnitt. Beim Doppelpendel ist der Phasenraum vierdimensional und die Koordinaten eines Punktes haben wir mit $(\varphi_1, \dot{\varphi}_1, \varphi_2, \dot{\varphi}_2)$ bezeichnet. Während der Bewegung des Doppelpendels wandert dieser Punkt im Phasenraum auf einer Trajektorie, also einer Kurve im Phasenraum.

Wenn man nun eine Hyperebene (das ist ein $(n-1)$ - dimensionaler Unterraum des Phasenraumes) in den Phasenraum legt, kann man die Punkte untersuchen, in denen die Trajektorie der Bahn diese Hyperebene schneidet.

In unserem Fall wollen wir die Hyperebene definieren durch die Bedingung $\varphi_1 = 0$. Wir betrachten also die Punkte im Phasenraum zum Zeitpunkt, wenn das obere Pendel gerade durch die y-Achse geht. Dadurch erhalten wir ein *diskretes* dynamisches System für das Doppelpendel. Ein Schnittpunkt

mit der Hyperebene wird abgebildet auf den Punkt, in welchem die Trajektorie die Hyperebene beim nächsten Durchgang schneidet:

$$(0, \dot{\varphi}_1, \varphi_2, \dot{\varphi}_2)_n \mapsto (0, \dot{\varphi}_1, \varphi_2, \dot{\varphi}_2)_{n+1}$$

Die Gesamtenergie entlang einer Trajektorie ist konstant. Für $\varphi_1 = 0$ ist sie mit den bisherigen Bezeichnungen und $m_1 = 1$, $m_2 = M$ gegeben durch:

$$\begin{aligned} E_{tot} &= \frac{1+M}{2} l_1^2 \dot{\varphi}_1^2 + M l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \cos \varphi_2 + \frac{M}{2} l_2^2 \dot{\varphi}_2^2 + g((1-l_1)(1+M) - M l_2 \cos \varphi_2) \\ &= \frac{1+M}{2} l_1^2 \dot{\varphi}_1^2 + M l_1 l_2 \dot{\varphi}_1 \dot{\varphi}_2 \cos \varphi_2 + \frac{M}{2} l_2^2 \dot{\varphi}_2^2 + E_{pot}(0, \varphi_2) \end{aligned}$$

Bei der Implementierung werden wir die Stelle $\varphi_1 = 0$ durch einen Vorzeichenwechsel des entsprechenden Parameters u_1 im Runge Kutta-Verfahren charakterisieren.

Nun projizieren wir die Hyperebene $(\dot{\varphi}_1, \varphi_2, \dot{\varphi}_2)$ auf $(\varphi_2, \dot{\varphi}_2)$, das heisst, wir tragen nur die Koordinaten des unteren Pendels in den Phasenraum ein. Wenn $(\varphi_2, \dot{\varphi}_2)$ gegeben sind, ist nämlich auch $\dot{\varphi}_1$ durch die obige Energiegleichung bestimmt. Das heisst, $\dot{\varphi}_1$ ist kein unabhängiger Parameter, da die Energie konstant ist. Wenn wir die Energiegleichung etwas umformen, erhalten wir eine Gleichung für $\dot{\varphi}_1$:

$$\frac{1+M}{2} l_1^2 \dot{\varphi}_1^2 + M l_1 l_2 \dot{\varphi}_2 \cos \varphi_2 \dot{\varphi}_1 + \frac{M}{2} l_2^2 \dot{\varphi}_2^2 + E_{pot}(0, \varphi_2) - E_{tot} = 0$$

Wir dividieren durch den ersten Koeffizienten von $\dot{\varphi}_1$:

$$\dot{\varphi}_1^2 + 2 \cdot \frac{M}{1+M} \cdot \frac{l_2}{l_1} \dot{\varphi}_2 \cos \varphi_2 \cdot \dot{\varphi}_1 + C = 0$$

Wobei

$$C = \frac{1}{(1+M)l_1^2} (M l_2^2 \dot{\varphi}_2^2 + 2E_{pot}(0, \varphi_2) - 2E_{tot})$$

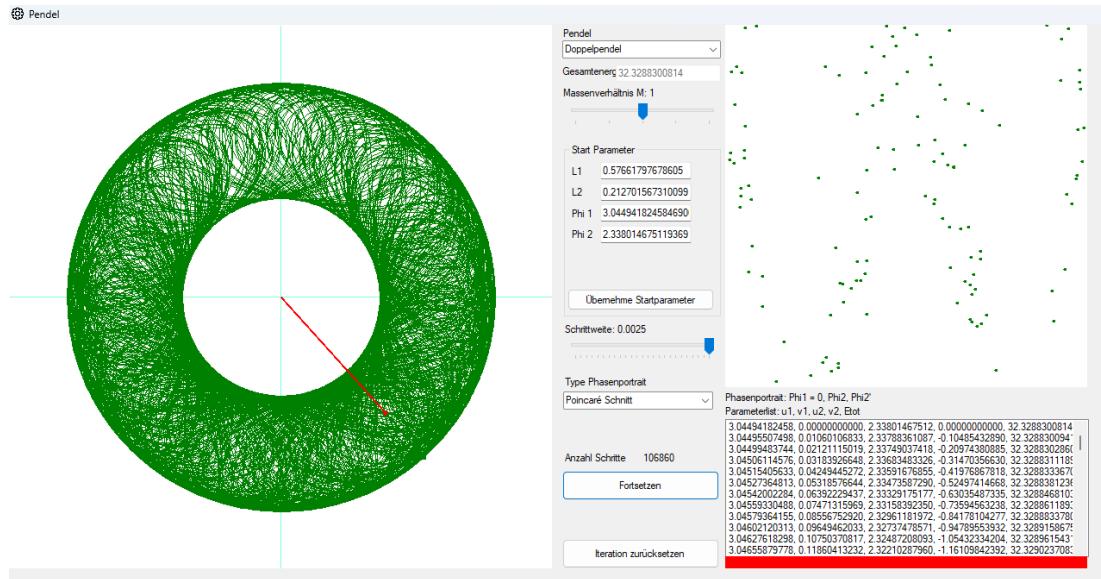
Das liefert für $\dot{\varphi}_1$:

$$\dot{\varphi}_1 = - \frac{M l_2}{(1+M)l_1} \dot{\varphi}_2 \cos \varphi_2 \pm \sqrt{\left(\frac{M l_2}{(1+M)l_1} \dot{\varphi}_2 \cos \varphi_2\right)^2 - C}$$

Bei der Implementierung werden wir zusätzlich als Bedingung für das Plotten von $(\varphi_2, \dot{\varphi}_2)$ verlangen, dass gilt:

$$\dot{\varphi}_1 + \frac{M}{1+M} \frac{l_2}{l_1} \dot{\varphi}_2 \cos \varphi_2 \geq 0$$

Damit die Punkte $(0, \dot{\varphi}_1, \varphi_2, \dot{\varphi}_2)$ eindeutig definiert sind.

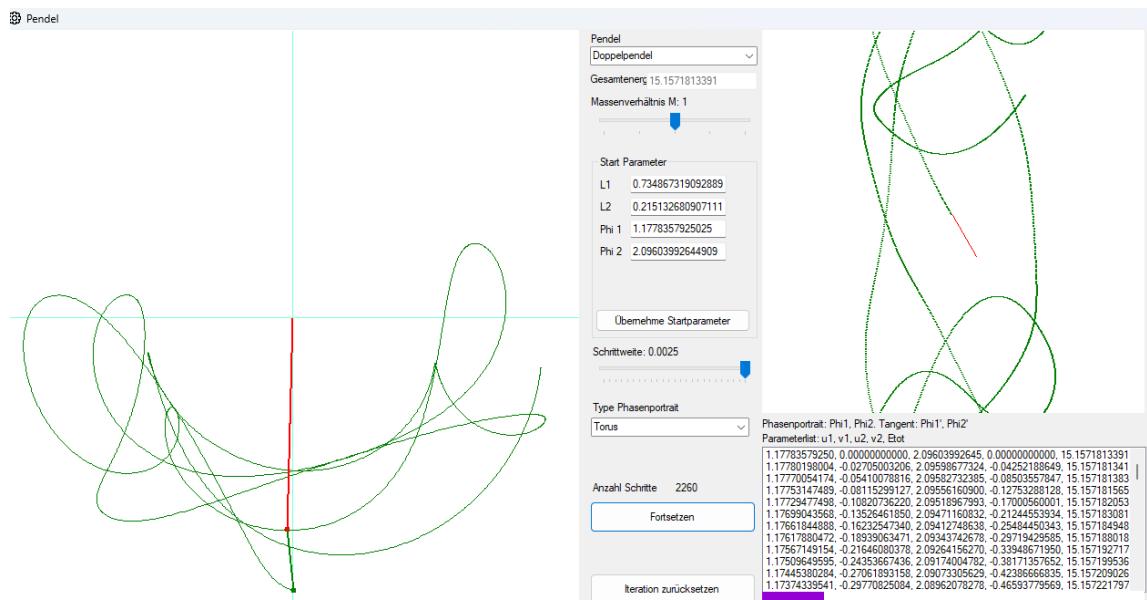


Poincaré Schnitt: Zu beachten ist, dass die Simulation nicht ein wirkliches Doppelpendel wiedergibt

In den Runge Kutta-Parametern lautet die Bedingung:

$$v_1 + \frac{M}{1+M} \frac{l_2}{l_1} v_2 \cos u_2 \geq 0$$

Eine weitere Darstellung des Phasenraumes erhält man durch den Torus. Die Parameter (φ_1, φ_2) sind 2π -periodisch und liegen auf einem Torus. Deren Ableitung im Punkt (φ_1, φ_2) ist die Tangente an die entsprechende Kurve und liegt in der Tangentialebene in diesem Punkt. Bei der Darstellung auf dem Torus wird der Orbit des Parameterpaars (φ_1, φ_2) grün dargestellt und die jeweilige Tangente ($\dot{\varphi}_1, \dot{\varphi}_2$) durch eine rote Linie ausgehend vom Punkt (φ_1, φ_2).

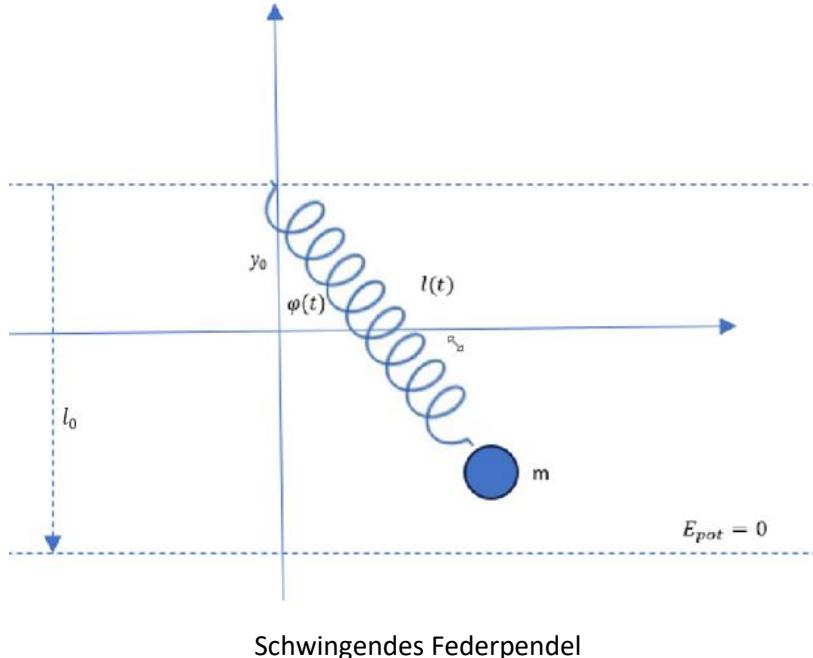


Darstellung der Bewegung auf dem Torus

In der *FrmPendulum* kann man dann wählen, welche Darstellung im Phasenraum (beide Pendel unabhängig voneinander, Torus oder Poincaré-Schnitt) man sehen möchte.

6.5. Schwingendes Federpendel

Hier betrachten wir ein Federpendel, welches zusätzlich wie ein Fadenpendel hin- und herschwingen kann.



Im Punkt $(0, y_0)$ des Koordinatensystems ist eine Feder befestigt, an deren anderem Ende eine Masse m hängt. Diese Masse schwingt zudem hin und her. Dabei ist der Ausschlagwinkel $\varphi(t)$ wie auch die Federlänge $l(t) > 0$ abhängig von der Zeit t .

Es sei l_0 die Länge der entspannten Feder. Das Nullniveau der potenziellen Energie in Bezug auf die Gravitation sei auf der Höhe $y = y_0 - l_0$ bzw. bei $\varphi = 0, l = l_0$. Mit D bezeichnen wir die Federkonstante.

Der Ortsvektor der Masse m ist:

$$\vec{r}(t) = l(t) \begin{pmatrix} \sin \varphi(t) \\ -\cos \varphi(t) \end{pmatrix} + \begin{pmatrix} 0 \\ y_0 \end{pmatrix}$$

$$\dot{\vec{r}} = l \begin{pmatrix} \sin \varphi \\ -\cos \varphi \end{pmatrix} + l \dot{\varphi} \begin{pmatrix} \cos \varphi \\ \sin \varphi \end{pmatrix}$$

$$|\dot{\vec{r}}|^2 = l^2 + 2l\dot{l}\dot{\varphi}[\sin \varphi \cos \varphi - \cos \varphi \sin \varphi] + l^2\dot{\varphi}^2$$

$$E_{kin} = \frac{1}{2}m(l^2 + l^2\dot{\varphi}^2)$$

Dann gilt für das gesamte Potenzial (Federenergie plus Gravitationsenergie):

$$V = \frac{1}{2}D(l - l_0)^2 + mg(l_0 - l \cos \varphi)$$

Damit haben wir die Lagrange-Funktion:

$$L = \frac{m}{2}(l^2 + l^2\dot{\varphi}^2) - \frac{D}{2}(l - l_0)^2 - mg(l_0 - l \cos \varphi)$$

Für die erste Koordinate $l(t)$ gilt:

$$\frac{d}{dt} \frac{\partial L}{\partial \ddot{l}} = \frac{d}{dt} (m\ddot{l}) = m\ddot{l}$$

$$\frac{\partial L}{\partial l} = ml\dot{\varphi}^2 - D(l - l_0) + mg\cos\varphi$$

Das liefert die Lagrange-Gleichung:

$$\ddot{l} = l\dot{\varphi}^2 - \frac{D}{m}(l - l_0) + g\cos\varphi$$

Für die zweite Koordinate $\varphi(t)$ hat man:

$$\begin{aligned} \frac{d}{dt} \frac{\partial L}{\partial \dot{\varphi}} &= \frac{d}{dt} (ml^2\dot{\varphi}) = 2ml\ddot{l}\dot{\varphi} + ml^2\ddot{\varphi} \\ \frac{\partial L}{\partial \varphi} &= -mglsin\varphi \end{aligned}$$

Das liefert die Lagrange-Gleichung:

$$\begin{aligned} ml^2\ddot{\varphi} &= -2ml\ddot{l}\dot{\varphi} - mglsin\varphi \\ \ddot{\varphi} &= -(2\dot{\varphi}\ddot{l} + gsin\varphi)/l \end{aligned}$$

Relevant für die Bewegung ist nur das Verhältnis $\frac{D}{m}$. Das ist aber gerade das Quadrat der Schwingungsfrequenz des Federpendels. Wir setzen also $\omega = \sqrt{\frac{D}{m}}$ und erhalten die Gleichungen:

$$\begin{cases} \ddot{l} = l\dot{\varphi}^2 - \omega^2(l - l_0) + g\cos\varphi \\ \ddot{\varphi} = -\frac{2\dot{\varphi}\ddot{l} + gsin\varphi}{l} \end{cases}$$

Damit wir die Gleichungen in ein Differentialgleichungssystem erster Ordnung überführen können, führen wir folgende Variablen ein:

$$u_1 = l, v_1 = \dot{l}, u_2 = \varphi, v_2 = \dot{\varphi}$$

Damit erhalten wir das Gleichungssystem für das Runge Kutta-Verfahren:

$$\begin{cases} \dot{u}_1 = v_1 \\ \dot{v}_1 = u_1 v_2^2 - \omega^2(u_1 - l_0) + g\cos u_2 \\ \dot{u}_2 = v_2 \\ \dot{v}_2 = -\frac{2v_2 v_1 + g\sin u_2}{u_1} \end{cases}$$

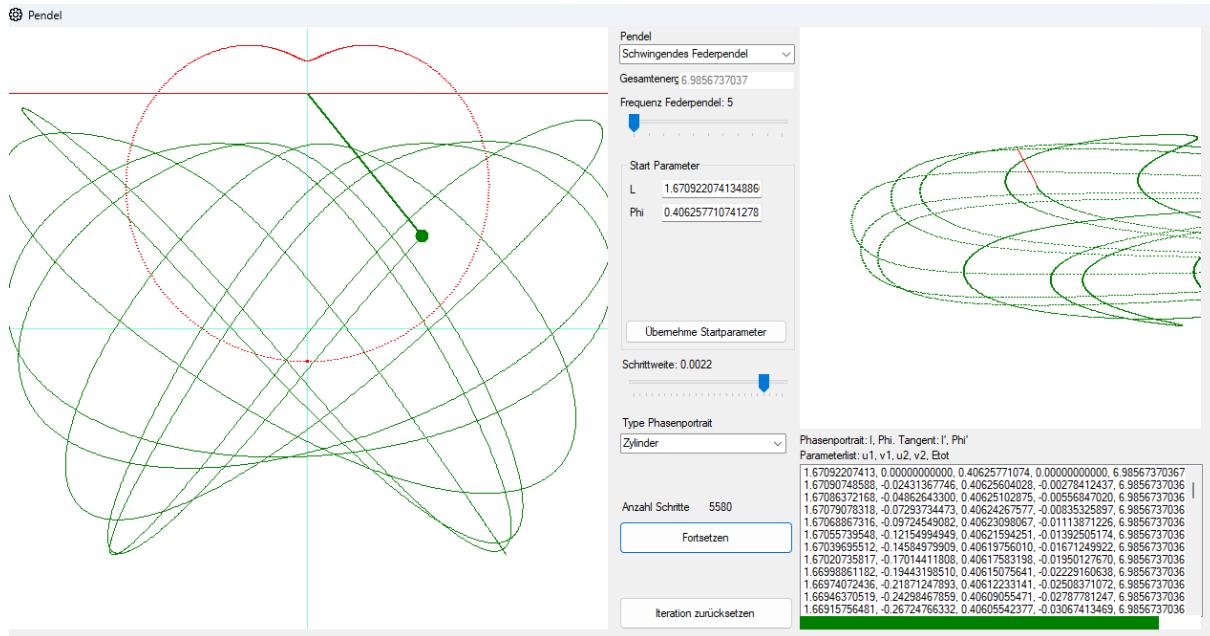
Bei der Implementierung wird derselbe Formalismus verwendet, wie beim Doppelpendel.

Da u_1 , also die Länge des Federpendels, im Nenner der letzten Gleichung steht, kann \dot{v}_2 sehr gross werden, wenn das Pendel in Richtung des Punktes $(0, y_0)$ schwingt und diesem sehr nahekommt. Bei einem physischen Pendel blockiert dann die Feder, deren Länge ein Mindestmass hat, diesen Effekt. Wir nehmen also bei der Implementierung an, dass die Länge des Federpendels einen Wert l_{min} nicht unterschreiten soll. Dieser Wert wird bei der Implementierung festgelegt.

Wenn nun während der Iteration $u_1 < l_{min}$ wird, dann wird das aufgefangen und an Stelle der Runge Kutta-Formeln $u_1 = l_{min}, v_1 = -v_1$ gesetzt. Die Gesamtenergie ändert sich dabei höchstens

unwesentlich. Optisch sieht das aus, wie wenn die Masse m an der Kugel mit Radius l_{min} und Mittelpunkt $(0, y_0)$ reflektiert wird, und zwar reibungsfrei.

Die Bedienung der *FrmPendulum* ist wieder dieselbe.



Schwingendes Federpendel

Für einen beliebigen Winkel φ gibt es einen Startpunkt des Pendels, an welchem die Federkraft gerade die Gravitationskraft kompensiert. In einer Übung kann der Leser zeigen, dass dies der Fall ist, wenn $l(\varphi) = l_0 + \frac{g \cos \varphi}{\omega^2}$. Die entsprechende Ortslinie ist eine Kardioide und im obigen Bild rot eingezeichnet.

Für die Darstellung im Phasenraum hat man wieder drei Möglichkeiten:

- Bewegung des Federpendels und der ebenen Schwingung unabhängig voneinander
- Darstellung des Parameterpaars (l, φ) auf einem Zylinder und der Tangente $(\dot{l}, \dot{\varphi})$ als rote Linie im jeweiligen aktuellen Kurvenpunkt
- Poincaré Schnitt: Hier wird die Hyperebene durch die Bedingung $\varphi = 0$ definiert, also an der Stelle, bei der das Pendel die negative y-Achse passiert.

Das Runge Kutta-Verfahren scheint hier zuverlässiger. Mindestens bleibt die Gesamtenergie des Systems erhalten.

6.6. Implementierung des schwingenden Federpendels

Die Implementierung folgt derselben Logik wie bereits die Implementierung des Doppelpendels. Implementiert werden die entsprechenden Runge Kutta-Formeln. Zusätzlich sollen die Startbedingungen für das Pendel möglichst so eingegrenzt werden, dass die Bewegung innerhalb des *PicDiagram* sichtbar ist.

Als einen Testfall für die Implementierung wollen wir den Gleichgewichtszustand betrachten, wenn $\varphi = 0$ ist. Die Länge der Feder sei in dieser Position l_1 .

In diesem Fall gilt:

$$D(l_1 - l_0) = mg$$

Bei der Implementierung setzen wir $m = 1$, dann ist $D = \omega^2$ und somit:

$$l_1 = l_0 + \frac{g}{\omega^2}$$

Wobei in der Implementierung $g = 9.81$ gesetzt wird.

Die Gesamtenergie, welche im zweitobersten Feld angezeigt wird, ist dann

$$E_{0,tot} = E_{kin} + E_{Feder} + E_{pot} = 0 + \frac{D}{2}(l_1 - l_0)^2 + mg(l_0 - l_1) = -\frac{g^2}{2\omega^2}$$

Wenn man die Gleichgewichtsbedingung und die Werte für die Implementierung beachtet. Man kann nun die entsprechenden Werte manuell berechnen, die Startwerte entsprechend setzen und übernehmen lassen. Dann kann man die angezeigte Gesamtenergie kontrollieren und zudem sollte sich das Pendel nicht bewegen.

Analog kann man den oberen Gleichgewichtspunkt betrachten: $\varphi = \pi$ und die Länge l_2 der Feder ist gerade so gross, dass sie die Schwerkraft kompensiert. Dann findet man nach einer analogen Rechnung:

$$l_2 = l_0 - \frac{g}{\omega^2}$$

Und die Energie in diesem Punkt ist:

$$E_{1,tot} = \frac{g^2}{2\omega^2}$$

Wenn wir die Gesamtenergie des Systems untersuchen, wobei das Nullniveau der gravitationsbedingten potenzielle Energie wieder auf der Höhe $y = y_0 - l_0$ ist, dann gilt für die Energie (wie vorhin ist $m = 1, D = \omega^2$):

$$\begin{aligned} E_{tot} &= E_{kin} + E_{Feder} + E_{pot} = \frac{1}{2}m(l^2 + l^2\dot{\varphi}^2) + \frac{D}{2}(l - l_0)^2 + mg(l_0 - l\cos\varphi) \geq \\ &\quad \frac{\omega^2}{2}(l - l_0)^2 + g(l_0 - l) =: \tilde{E}(l) \end{aligned}$$

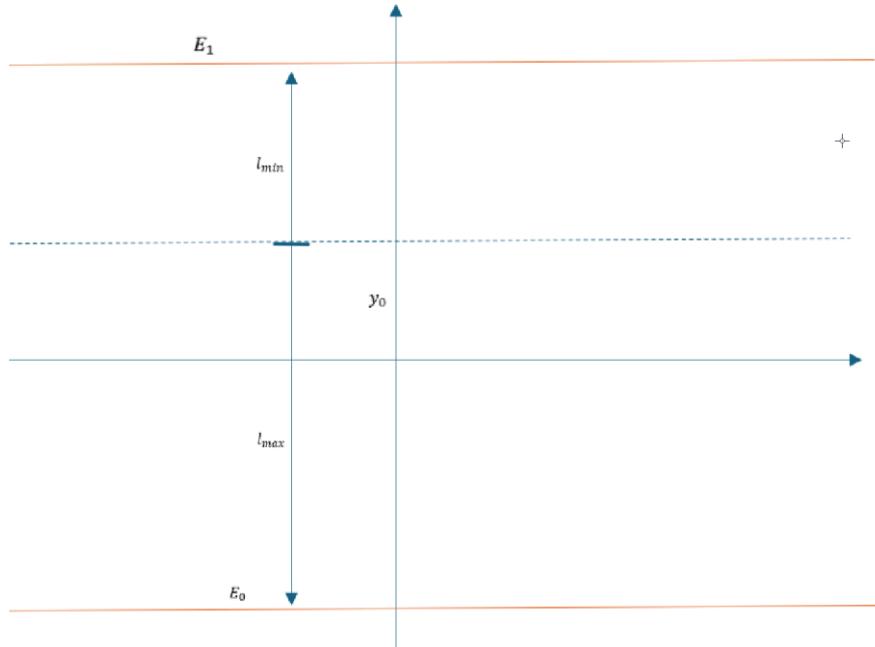
$\tilde{E}(l)$ ist eine nach oben offene Parabel und wir suchen deren Minimum:

$$\frac{d\tilde{E}}{dl} = \omega^2(l - l_0) - g = 0$$

Das Minimum wird gerade im unteren Gleichgewichtszustand angenommen: $l_{min} = l_0 + \frac{g}{\omega^2}$ und damit gilt:

$$E_{tot} \geq -\frac{g^2}{2\omega^2}, \forall l, \varphi$$

Bei der Implementierung müssen wir abklären, wie die Parameter l_0, y_0, ω gewählt werden sollen, damit das Pendel bei seinen Schwingungen im Quadrat $[-1,1]x[-1,1]$ Platz hat.



Analyse der maximalen Pendelbewegung

Zuerst kümmern wir uns um einen optimalen Wert von y_0 .

Nehmen wir an, das Pendel startet aus der unteren Position und hat dort eine maximale Länge l_{max} und es sei $\varphi = 0$. Die Pendelenergie in diesem Punkt sei E_0 . Auf der maximalen Höhe, welche das Pendel erreicht, ist die Pendellänge minimal l_{min} . Die Energie auf dieser Höhe sei E_1 . Insgesamt steht in vertikaler Richtung die Länge 2 zur Verfügung, d.h. $l_{max} + l_{min} = 2$.

Wegen der Erhaltung der Energie gilt dann $E_0 = E_1$. Weil wir bei der Implementierung m=1 setzen und im oberen Punkt $\varphi = \pi$ ist ergibt das:

$$\frac{\omega^2}{2}(l_{max} - l_0)^2 + g(l_0 - l_{max}) = \frac{\omega^2}{2}(l_{min} - l_0)^2 + g(l_0 + l_{min})$$

Nach kurzer Rechnung erhalten wir daraus:

$$\frac{\omega^2}{2}(l_{max}^2 - l_{min}^2 - 2L(l_{max} - l_{min})) - g(l_{max} + l_{min}) = 0$$

Nun verwenden wir: $l_{max} + l_{min} = 2$ und erhalten:

$$\frac{\omega^2}{2}(2(l_{max} - l_{min}) - 2L(l_{max} - l_{min})) = 2g$$

$$l_{max} - l_{min} = \frac{2g}{\omega^2(1 - L)}$$

Der Nenner ist definiert, da $L < 1$. Nun ist

$$2y_0 = l_{max} - l_{min}$$

Damit ist der Aufhängepunkt des Pendels genügend gut platziert, wenn gesetzt wird:

$$y_0 = \frac{g}{\omega^2(1 - l_0)}$$

Da gilt: $y_0 \leq 1$ lässt sich daraus noch eine Bedingung für ω ableiten:

$$\frac{g}{\omega^2(1 - l_0)} \leq 1 \Rightarrow \sqrt{\frac{g}{1 - l_0}} \leq \omega$$

Alle diese Bedingungen werden bei der Implementierung beachtet.

Zusätzlich soll der obere Gleichgewichtspunkt «möglich» sein. Für diesen gilt, wie wir weiter oben gesehen haben:

$$l_2 = l_0 - \frac{g}{\omega^2}$$

Da $l_2 > 0$ sein soll, folgt $l_0 - \frac{g}{\omega^2} > 0$ und damit $\omega > \sqrt{\frac{g}{l_0}}$. Zusammenfassend gilt:

$$\omega > \max\left(\sqrt{\frac{g}{l_0}}, \sqrt{\frac{g}{1 - l_0}}\right)$$

Das legt aus Symmetriegründen nahe, bei der Implementierung $l_0 = 0.5$ zu wählen.

Damit wir die Gesamtenergie in der Statusbar rechts unten realistisch anzeigen können, benötigen wir noch die Bandbreite, in welcher sich die Energie beim Start liegen kann. Zur Überwachung der aktuellen Energie im Laufe der Pendelbewegung wird diese dann in einem Intervall $[E_{min}, E_{max}]$ in der Statusbar unten rechts im Fenster dargestellt. Wir haben bereits festgestellt, dass gilt:

$$E_{tot} \geq -\frac{g^2}{2\omega^2} =: E_{min}$$

Die Gesamtenergie am Start ist:

$$E_{tot} = \frac{\omega^2}{2}(l - l_0)^2 + g(l_0 - l \cos \varphi)$$

Also eine nach oben geöffnete Parabel. Diese nimmt den maximalen Wert an Rande des Definitionssintervalls an. Für die Pendellänge haben wir bei der Implementierung festgelegt:

$$l \in [l_{min}, 1 + y_0 \cos \varphi]$$

Der maximale Wert von l ist also abhängig von φ und ist gleich: $l(\varphi) = 1 + y_0 \cos \varphi$. Somit ist

$$E_{tot}(\varphi) = \frac{\omega^2}{2}(1 + y_0 \cos \varphi - l_0)^2 + g(l_0 - (1 + y_0 \cos \varphi) \cos \varphi)$$

Wir suchen nun das Extremum dieser Funktion. Wie man nachrechnen kann, ist

$$\frac{dE_{tot}}{d\varphi} = 0 \text{ für } \varphi = 0, \pi.$$

Berechnet man die Differenz der Energiewerte an diesen Stellen, erhält man:

$$E_{tot}(\pi) - E_{tot}(0) =$$

$$\frac{\omega^2}{2}(1 - y_0 - l_0)^2 + g(l_0 + 1 - y_0) - \frac{\omega^2}{2}(1 + y_0 - l_0)^2 + g(l_0 - 1 - y_0) =$$

$$2y_0 \omega^2 (l_0 - 1) + 2g \geq 0$$

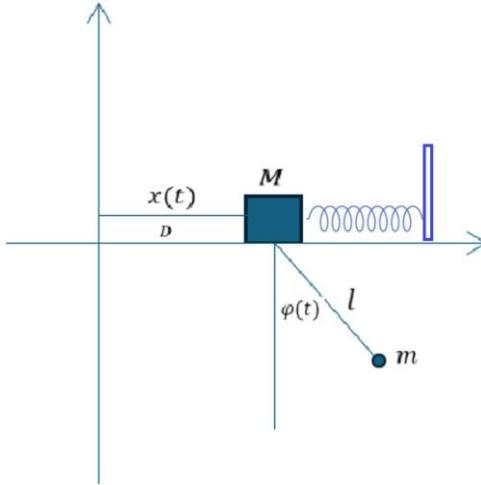
Denn wir haben gesetzt:

$$y_0 = \frac{g}{\omega^2(1 - l_0)}$$

Somit ist $E_{max} = E_{tot}(\pi)$.

6.7. Rüttelpendel

In diesem Abschnitt betrachten wir ein Rüttelpendel, bei welchem eine Masse M horizontal wie ein Federpendel reibungsfrei hin- und herschwingt und an welcher ein Fadenpendel befestigt ist.



Rüttelpendel

Der Ortsvektor der Masse m ist:

$$\vec{r}(t) = \begin{pmatrix} x(t) + l \sin \varphi(t) \\ -l \cos \varphi(t) \end{pmatrix}$$

Damit erhalten wir:

$$\dot{\vec{r}} = \begin{pmatrix} \dot{x} + l \dot{\varphi} \cos \varphi \\ l \dot{\varphi} \sin \varphi \end{pmatrix}$$

$$|\dot{\vec{r}}|^2 = \dot{x}^2 + 2\dot{x}\dot{\varphi}l \cos \varphi + \dot{\varphi}^2 l^2$$

$$E_{kin} = \frac{1}{2}m[\dot{x}^2 + 2\dot{x}\dot{\varphi}l \cos \varphi + \dot{\varphi}^2 l^2] + \frac{1}{2}M\dot{x}^2$$

D sei die Federkonstante. Die Feder sei entspannt an der Stelle $x = 0$. Das Nullniveau der potenziellen Energie der Masse m sei auf dem Niveau $y = 0$. Dann ist:

$$V = \frac{1}{2}Dx^2 - mg l \cos \varphi$$

Und die Lagrange-Funktion:

$$L = \frac{1}{2}m[\dot{x}^2 + 2\dot{x}\dot{\varphi}l \cos \varphi + \dot{\varphi}^2 l^2] + \frac{1}{2}M\dot{x}^2 - \frac{1}{2}Dx^2 + mg l \cos \varphi$$

Für die Koordinate φ liefert das:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\varphi}} = \frac{d}{dt} (m\dot{x}l \cos \varphi + m\dot{\varphi}l^2) = ml(\ddot{x} \cos \varphi - \dot{x} \dot{\varphi} \sin \varphi + \ddot{\varphi}l)$$

$$\frac{\partial L}{\partial \varphi} = -m\dot{x}\varphi l \sin\varphi - mg l \sin\varphi = -ml(\dot{x}\varphi + g)\sin\varphi$$

Das liefert die erste Gleichung:

$$ml(\ddot{x}\cos\varphi - \dot{x}\varphi \sin\varphi + \dot{\varphi}l) = -ml(\dot{x}\varphi + g)\sin\varphi$$

$$\ddot{x}\cos\varphi + \dot{\varphi}l + g\sin\varphi = 0$$

Für die Koordinate x erhalten wir:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{x}} = \frac{d}{dt} (m\dot{x} + \dot{\varphi}l \cos\varphi + M\dot{x}) = (m+M)\ddot{x} + \dot{\varphi}l \cos\varphi - \dot{\varphi}^2 l \sin\varphi$$

$$\frac{\partial L}{\partial x} = -Dx$$

Das liefert die zweite Gleichung:

$$(m+M)\ddot{x} + \dot{\varphi}l \cos\varphi - \dot{\varphi}^2 l \sin\varphi + Dx = 0$$

Wenn wir die erste Gleichung mit $\cos\varphi$ multiplizieren und die Differenz der Gleichungen bilden, erhalten wir:

$$\ddot{x}(\cos^2\varphi - (m+M)) + g\sin\varphi \cos\varphi + \dot{\varphi}^2 l \sin\varphi - Dx = 0$$

Bei der Implementierung werden wir dafür sorgen, dass $m+M > 1$ ist. Dann gilt:

$$\ddot{x} = \frac{\dot{\varphi}^2 l \sin\varphi + g\sin\varphi \cos\varphi - Dx}{m+M - \cos^2\varphi}$$

Wir können dieses Resultat für \ddot{x} in die erste Gleichung einsetzen und erhalten nach kurzer Umformung:

$$\ddot{\varphi} = \frac{Dx \cos\varphi - (m+M)g\sin\varphi - \dot{\varphi}^2 l \sin\varphi \cos\varphi}{l(m+M - \cos^2\varphi)}$$

Um ein Differentialgleichungssystem erster Ordnung zu erhalten, setzen wir:

$$u_1 = x, v_1 = \dot{x}, u_2 = \varphi, v_2 = \dot{\varphi}$$

Damit erhalten wir das Gleichungssystem für das Runge Kutta-Verfahren:

$$\left\{ \begin{array}{l} \dot{u}_1 = v_1 \\ \dot{v}_1 = \frac{v_2^2 l \sin u_2 + g \sin u_2 \cos u_2 - Du_1}{m+M - \cos^2 u_2} \\ \dot{u}_2 = v_2 \\ \dot{v}_2 = \frac{Du_1 \cos u_2 - (m+M)g \sin u_2 - v_2^2 l \sin u_2 \cos u_2}{l(m+M - \cos^2 u_2)} \end{array} \right.$$

Bei der Implementierung wird l durch die manuelle Positionierung der Masse m bestimmt. Ferner normieren wir in der Implementierung $m = 1$. Die Wahl von D, M wird später diskutiert.

Für die Implementierung werden wir die Gesamtenergie des Systems brauchen. Da beim Start gilt $\dot{x} = 0, \dot{\varphi} = 0$, ist diese nur potenziell:

$$E_{Start} = \frac{1}{2} D x^2 - m g l \cos\varphi$$

Deren Range ist: $E_{min} = -mgl$, $E_{max} = \frac{D}{2} + mgl$, da $x \in [-1,1]$, $\varphi \in [-\pi, \pi]$.

Die Energie zu einem beliebigen Zeitpunkt ist:

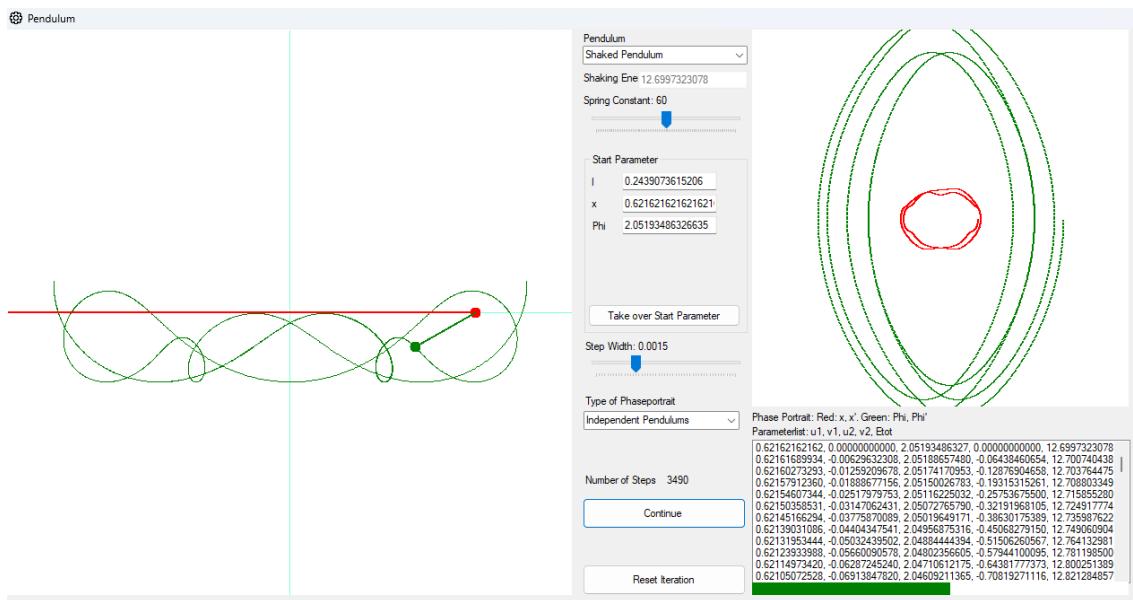
$$E_{tot} = \frac{1}{2}m[\dot{x}^2 + 2\dot{x}\dot{\varphi}l\cos\varphi + \dot{\varphi}^2l^2] + \frac{1}{2}M\dot{x}^2 + \frac{1}{2}Dx^2 - mgl\cos\varphi$$

Bei der Implementierung ist zudem l so zu wählen, dass das Fadenpendel sichtbar bleibt. Aus Symmetriegründen genügt es, den Fall $x > 0$ zu betrachten. Wenn auch $\varphi > 0$ ist, dann soll gelten:

$$x + l\sin\varphi < 1$$

Wenn $x < 0$ und $\varphi < 0$ gilt analog: $|x| + l|\sin\varphi| < 1$. Das ist in der Implementierung so umgesetzt.

Nach einigem Experimentieren wurde bei der Implementierung $M = 2$ festgelegt und die Federkonstante D als zusätzlicher Parameter variabel wählbar gemacht.



Rüttelpendel

6.8. Übungsbeispiele

- Untersuche beim schwingenden Federpendel für einen beliebigen Winkel φ , für welche Werte von $l(\varphi)$ die Feder gerade die Gravitationskraft aufhebt.
- Angenommen, man hat ein Doppelpendel und es gilt: $l_1 = l_2 = 1/2$, $m_1 = m_2 = 1$. Am Start sei $\varphi_1 = \varphi_2 > 0$. Gilt dann auch nach weiteren Iterationsschritten $\varphi_1 = \varphi_2$? (Untersuche die entsprechenden Formeln des Runge Kutta-Verfahrens.)
- Experimentiere mit dem «Simulator» und untersche, für welche Anfangsbedingungen bei jedem Pendel die Gesamtenergie während der Pendelbewegung einigermassen konstant bleibt.
- Untersche ein Rüttelpendel, bei dem das Federpendel in vertikaler Richtung auf und ab schwingt. An diesem Federpendel ist ein Fadenpendel befestigt.
 - Berechne die Lagrange Funktion für dieses Pendel
 - Leite damit die Differentialgleichungen her, welche die Bewegung beschreiben

- c) Führe diese Differentialgleichungen durch geeignete Substitution über in ein System von Differentialgleichungen erster Ordnung
- d) Erstelle daraus die Iterationsgleichungen für das Runge Kutta-Verfahren
- e) Die Pendel sollen aus einer Ruhelage starten, das heisst, dass beim Start nur potenzielle Energie vorhanden ist und die kinetische Energie am Start Null ist. In welchem Intervall liegt die Gesamtenergie des Systems?
- f) Implementiere dieses Pendel im «Simulator»

7. Iteration in der komplexen Ebene

Dieses Kapitel ist in Bearbeitung!

7.1. Einführung

7.2. Das Newton Verfahren

Das Newton Verfahren für reelle Funktionen ist meist Bestandteil des Unterrichts an der Mittelschule. Wir wollen hier nur eine kurze Zusammenfassung geben und darauf verzichten, das Newton Verfahren im Reellen didaktisch ausführlich herzuleiten oder graphisch zu motivieren.

Es sei $f(x) \in C^2[a, b]$ eine zweimal stetig differenzierbare reelle Funktion auf einem Intervall $[a, b]$. Wir nehmen ferner an, dass f eine Nullstelle in diesem Intervall besitzt, d.h. $\exists \xi \in [a, b] \text{ mit } f(\xi) = 0$. Ferner soll für $x \in [a, b]$ gelten: $f'(x) \neq 0$.

Wir definieren nun die Funktion:

$$g(x) := x - \frac{f(x)}{f'(x)}, [a, b] \rightarrow \mathbb{R}$$

Satz

- a) ξ ist ein Fixpunkt von $g \Leftrightarrow \xi$ ist eine Nullstelle von f
- b) Sei ξ ein Fixpunkt von $g \Rightarrow \xi$ ist attraktiv

Beweis:

- a) $g(\xi) = \xi \Leftrightarrow \xi - \frac{f(\xi)}{f'(\xi)} = \xi \Leftrightarrow f(\xi) = 0$
- b) Wir untersuchen gemäss Kapitel 2.3 die Ableitung: $|g'(\xi)| = \left| 1 - \frac{f'(\xi)^2 - f(\xi)f''(\xi)}{f'(\xi)^2} \right| = 0 < 1$

□

Beim Newton Verfahren wird – ausgehend von einem Startpunkt x_1 nahe bei ξ – die Funktion g iteriert:

$$x_{n+1} = g(x_n) = x_n - \frac{f(x_n)}{f'(x_n)}$$

Es gibt also eine Umgebung

Es sei ferner: $m := \min_{x \in [a, b]} |f'(x)| > 0$ und $M := \max_{x \in [a, b]} |f''(x)|$. Wir betrachten eine Umgebung $U_\rho(\xi) \subset [a, b]$ mit $\rho < \frac{2m}{M}$.

Behauptung:

- c) $g: U_\rho(\xi) \rightarrow U_\rho(\xi)$, das heisst, g bildet $U_\rho(\xi)$ wieder auf $U_\rho(\xi)$ ab.
- d) ξ ist ein Fixpunkt von $g \Leftrightarrow \xi$ ist eine Nullstelle von f

Beweis:

- a) Wir betrachten $x \in U_\rho(\xi)$ und o.B.d.A. $x < \xi$. Die Taylorentwicklung von f im Punkt x mit dem Restglied von Lagrange lautet:

$$0 = f(\xi) = f(x) + f'(x)(\xi - x) + \frac{1}{2}f''(\eta)(\xi - x)^2, \eta \in [x, \xi] \quad (*)$$

Da $|f'(x)| \neq 0$ in $[a, b]$ gilt dann:

$$\left| -\frac{f(x)}{f'(x)} - \xi + x \right| = \frac{1}{2} \frac{|f''(\eta)|}{|f'(x)|} (\xi - x)^2 < \frac{M}{2m} (\xi - x)^2 < \frac{1}{\rho} \rho^2 = \rho$$

Somit ist:

$$|g(x) - \xi| = \left| x - \frac{f(x)}{f'(x)} - \xi \right| < \rho$$

b) $g(\xi) = \xi \Leftrightarrow \xi - \frac{f(\xi)}{f'(\xi)} = \xi \Leftrightarrow f(\xi) = 0 \quad \square$

Wir wählen nun einen Startwert x_1 der nahe genug bei ξ liegt, insbesondere sei $x_1 \in U_\rho(\xi)$. Dann iterieren wir die Funktion g ausgehend von x_1 :

$$x_{n+1} = g(x_n) = x_n - \frac{f(x_n)}{f'(x_n)}$$

Dies ist gerade das Newtonverfahren.

Dann gilt:

Satz

Sei die Folge (x_n) definiert durch die Iteration $x_{n+1} = g(x_n)$ mit Startwert $x_1 \in U_\rho(\xi)$, wobei ξ ein Fixpunkt von g ist.

- a) ξ ist attraktiv
- b) Die Folge (x_n) konvergiert gegen ξ

Beweis:

- a) Wir untersuchen (gemäss Kapitel 2.3) den Betrag der Ableitung:

$$|g'(\xi)| = \left| 1 - \frac{f'(\xi)^2 - f(\xi)f''(\xi)}{f'(\xi)^2} \right| = 0 < 1$$

- b) Unter Verwendung der Taylorentwicklung von vorhin erhalten wir:

$$|x_{n+1} - \xi| = |g(x_n) - \xi| = \frac{1}{2} \frac{|f''(\eta)|}{|f'(x)|} (x_n - \xi)^2 < \frac{M}{2m} (x_n - \xi)^2$$

Es ist $q := \frac{\rho M}{2m} < 1$ und $|x_n - \xi| < \rho$. Daraus folgt:

$$|x_{n+1} - \xi| < q|x_n - \xi| < q^{n-1}|x_1 - \xi| \rightarrow 0, n \rightarrow \infty$$

\square

Die Herausforderung ist natürlich, einen geeigneten Startwert $x_1 \in U_\rho(\xi)$ zu finden.

Möglicherweise wird die Taylorentwicklung mit Restglied von Lagrange in der Mittelschule nicht behandelt. Wir brauchen das auch nicht in dieser Allgemeinheit. Deshalb folgt hier ein Einschub, um die Abschätzung (*) direkt herzuleiten. Wir setzen aber den Satz von Rolle voraus. Damit beweisen wir den

Verallgemeinerter Mittelwertsatz

Seien g, h zwei stetige Funktionen auf einem Intervall $[a, b]$, welche auf $]a, b[$ differenzierbar sind. Ferner sei $h(x) \neq 0$ für $x \in]a, b[$.

Behauptung: Dann gibt es einen Punkt $\xi \in]a, b[$ so dass

$$\frac{g(b) - g(a)}{h(b) - h(a)} = \frac{g'(\xi)}{h'(\xi)}$$

Beweis:

Betrachte die Hilfsfunktion

$$k(x) := (g(b) - g(a))h(x) - (h(b) - h(a))g(x)$$

Dann gilt wie man leicht nachrechnet:

$$k(a) = h(a)g(b) - g(a)h(b) = k(b)$$

Nach dem Satz von Rolle gibt es dann ein $\xi \in]a, b[$ mit $k'(\xi) = 0$. Daraus folgt unmittelbar die Behauptung.

Der bekannte Mittelwertsatz folgt daraus direkt, wenn man $h(x) = x$ setzt. \square

Bei der Abschätzung (*) wollen wir den Fehler abschätzen, wenn wir eine in einem Intervall $[a, b]$ zweimal differenzierbare Funktion durch eine lineare Approximation annähern:

$$f(x) \approx f(a) + f'(a)(x - a)$$

Wir betrachten dazu die Differenzfunktion:

$$g(x) := f(x) - f(a) - f'(a)(x - a)$$

Dann gilt: $g(a) = 0, g'(a) = 0$ und $g''(x) = f''(x), x \in [a, b]$.

Nach dem Mittelwertsatz gibt es ein $\xi \in]a, x[$ so dass

$$\frac{g'(x) - g'(a)}{x - a} = g''(\xi) \quad (**)$$

Nun definieren wir die Funktion $h(x) := (x - a)^2$. Dann folgt nach dem verallgemeinterten Mittelwertsatz, dass es ein $\xi_1 \in]a, x[$ gibt mit:

$$\frac{g(x) - g(a)}{h(x) - h(a)} = \frac{g'(\xi_1)}{h'(\xi_1)} = \frac{g'(\xi_1)}{2(\xi_1 - a)}$$

Nun verwenden wir (**) an der Stelle $x = \xi_1$. Wegen $g'(a) = 0$ ist dann:

$$g'(\xi_1) = g''(\xi)(\xi_1 - a)$$

Für ein $\xi \in]a, \xi_1[\subset]a, b[$.

Damit wird:

$$\frac{g(x) - g(a)}{h(x) - h(a)} = \frac{g(x)}{(x - a)^2} = \frac{g''(\xi)(\xi_1 - a)}{2(\xi_1 - a)}$$

Damit haben wir die Abschätzung:

$$g(x) = f(x) - f(a) - f'(a)(x - a) = \frac{g''(\xi)}{2}(x - a)^2$$

Für ein $\xi \in]a, b[$.

Im Folgenden übertragen wir nun das Newton Verfahren auf die komplexe Ebene. Auch diese ist in der Regel Gegenstand des Unterrichts in der Mittelschule, mindestens bei mathematisch ausgerichteten Abteilungen. Hingegen geht die Untersuchung von komplexen Funktionen bzw. die Funktionentheorie weit über den Mittelschulstoff hinaus. Wir werden aber lediglich einige wenige Grundlagen brauchen. Eine vollständige und didaktisch gute Einführung in die Funktionentheorie mit historischen Anmerkungen findet man z.B. in [14]. Für uns nützlich ist das erste Kapitel über die Grundlagen der komplexen Differentialrechnung sowie allenfalls der Abschnitt über die Taylorreihe in Kapitel sieben. Hier wollen wir dennoch ein kurzer Abriss der nötigen Grundlagen für die Übertragung des Newton Verfahrens auf die komplexe Ebene geben.

Wir setzen voraus:

- Definition der komplexen Ebene \mathbb{C} und das Rechnen mit komplexen Zahlen $z \in \mathbb{C}$
- Komplexe Polynome und der Fundamentalsatz der Algebra
- Polarkoordinaten und die Definition von $e^{ix} = \cos x + i \sin x$
- Standardmetrik in \mathbb{C} : Sei $z = x + iy \in \mathbb{C}$ dann ist: $|z| := \sqrt{x^2 + y^2}$

Mit dieser Metrik ist definiert, was es bedeutet, dass eine Folge $(z_n) \in \mathbb{C}$ gegen einen Grenzwert $a \in \mathbb{C}$ konvergiert, nämlich:

$$\lim_{n \rightarrow \infty} z_n = a \Leftrightarrow \lim_{n \rightarrow \infty} |z_n - a| = 0$$

Man kann also Iterationen untersuchen der Form:

$$z_{n+1} = g(z_n), n \in \mathbb{N}$$

Wobei g ein komplexes Polynom (oder im allgemeinen eine komplexe Funktion) ist.

Eine komplexe Funktion ist gegeben durch eine Vorschrift:

$$f: z = x + iy \in \mathbb{C} \mapsto w = u(x, y) + iv(x, y) \in \mathbb{C}$$

Mit der Standardmetrik ist definiert, was es bedeutet, dass f in einem Punkt $c \in \mathbb{C}$ stetig ist. In diesem Fall existiert der Grenzwert:

$$\lim_{z \rightarrow c} f(z) = f(c)$$

Die Ableitung einer komplexen Funktion ist formal gleich definiert wie im reellen. Eine komplexe Funktion f ist (komplex) differenzierbar in einem Punkt $c \in \mathbb{C}$, wenn der limes existiert:

$$\lim_{z \rightarrow c} \frac{f(z) - f(c)}{z - c} =: f'(c)$$

Wenn eine komplexe Funktion in jedem Punkt ihres Definitionsbereiches $D_f \subseteq \mathbb{C}$ differenzierbar ist, wird dadurch eine Ableitungsfunktion $z \in D_f \mapsto f'(z)$ definiert.

Da die Definition der Ableitung formal gleich definiert wird, wie im reellen, folgen sämtliche Ableitungsregeln wie im reellen. Deren Herleitung läuft analog wie im reellen und hier wird darauf verzichtet.

Wir werden das Newton Verfahren auf Nullstellen von komplexen Polynomen anwenden. Das sind Polynome der Form:

$$p(z) = a_0 + a_1 z + a_2 z^2 + \cdots + a_n z^n, a_i \in \mathbb{C}$$

Diese sind in ganz \mathbb{C} stetig und differenzierbar und die Ableitung ist gegeben durch:

$$p'(z) = a_1 + 2a_2 z + \cdots + n a_n z^{n-1}$$

Das Studium der komplexen Funktionen ist ausserordentlich interessant und schön. Weiter darauf einzugehen würde den Rahmen dieses Dokumentes sprengen. Einige Resultate seien aber trotzdem genannt.

Wenn eine komplexe Funktion gegeben ist durch eine Funktionsvorschrift:

$$f: x + iy \mapsto u(x, y) + iv(x, y)$$

Wobei u und v reelle Funktionen sind. Die Frage ist dann, wie beliebig diese reellen Funktionen gewählt werden dürfen, damit f komplex differenzierbar ist. Die Antwort darauf lautet:

Dann ist f nur genau dann differenzierbar, wenn die reellwertigen Funktionen $u(x, y), v(x, y)$ partiell stetig differenzierbar sind und die sogenannten Cauchy-Riemann'schen Differentialgleichungen gelten:

$$u_x = v_y, u_y = -v_x$$

Eine weitere Frage ist, ob irgendeine reellwertige Funktion $u(x, y)$, welche im reellen zweimal partiell stetig differenzierbar ist, geeignet durch eine weitere reellwertige Funktion $v(x, y)$ ergänzt werden kann, sodass $f(x + iy) = u(x, y) + iv(x, y)$ komplex differenzierbar ist. Die Antwort darauf lautet, dass notwendigerweise gelten muss:

$$u_{xx} + u_{yy} = 0, v_{xx} + v_{yy} = 0$$

Funktionen $u(x, y)$ mit dieser Eigenschaft nennt man *harmonische Funktionen*.

Ein weiteres bemerkenswerte Resultat ist folgendes:

Wenn eine komplexe Funktion f *einmal* komplex differenzierbar in einem Bereich $D \subseteq \mathbb{C}$ ist, dann ist sie in diesem Bereich *unendlich oft* komplex differenzierbar.

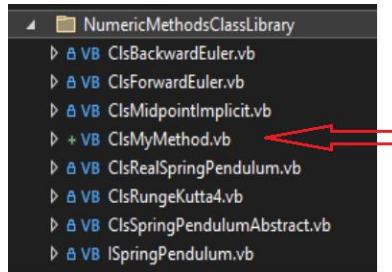
Nun kehren wir zurück zum Newton Verfahren.

8. Implementierung eigener Systeme im «Simulator»

Falls der Benutzer des «Simulator» eigener Varianten der bestehenden implementierten Systeme programmieren möchte, ist dies sehr einfach möglich. Wir wollen das am Beispiel einer eigener Variante einer numerischen Methode erläutern.

1. Schritt

Erstelle eine eigene Klasse *ClsMyMethod* und füge sie zum Ordner *NumericMethodsClassLibrary* hinzu:



Wichtig dabei ist, sich an die Konvention zu halten, dass Klassennamen mit dem Präfix *Cls* beginnen.

2. Schritt

Diese Klasse muss nun die Klasse *ClsSpringPendulumAbstract* übernehmen. Der einzige Codeteil, welcher gegenüber den bestehenden Klassen ändert, ist der Algorithmus der numerischen Methode:

```
0 Verweise
Public Class ClsMyMethod
    Inherits ClsSpringPendulumAbstract

    Private u As Decimal
    Private v As Decimal

    Protected Overrides Sub Iteration()

        Dim i As Integer

        With MyActualParameter
            For i = 1 To MyNumberOfApproxSteps
                .Component(0) += MyH

                'Component(1) holds the y-value
                u = .Component(1)
                v = .Component(2)

                'this is the numerical approximation
                .Component(1) = My own formula For the first component
                .Component(2) = My own formula For the second component
            Next

            'the Component(0) holds the "time" t with 2*pi period
            .Component(0) = .Component(0) Mod CDec(2 * Math.PI)
        End With
    End Sub
End Class
```

The code shows a VB.NET class named 'ClsMyMethod' that inherits from 'ClsSpringPendulumAbstract'. It contains two private decimal fields, 'u' and 'v'. The 'Iteration' method overrides the base class's implementation. Inside, it loops 'MyNumberOfApproxSteps' times, updating the first component of the state vector ('Component(0)'). It also updates the second component ('Component(2)'). A note in the code specifies that 'Component(1)' holds the y-value. The comments indicate that the numerical approximation formulas for both components are 'My own formula'. Finally, it uses the 'Mod' operator to keep the time component within a full period of 2*pi.

Der Code könnte dann wie oben gezeigt aussehen.

Weitere Schritte sind nicht nötig!! Insbesondere wird in der *FrmSpringPendulum* die Combobox mit der Auswahl der numerischen Methoden durch *Reflection* gefüllt. Da ist also keine Änderung nötig. In

der Combobox wird die eigene Methode als *MyMethod* aufgeführt. Das heisst, aus dem Namen wird das Präfix *Cl*s herausgeschnitten.

3. Schritt (optional)

Wenn man die Bezeichnungen in Deutsch und Englisch anders wählen will als *MyMethod*, kann man einen entsprechenden Schlüssel in den Resourcenfiles *LabelsEN* und *LabelsDE* eintragen. Das geschieht wie folgt:

In *LabelsEN*:

Cl <i>s</i> MyMethod	My own numeric method	Cl <i>s</i>
----------------------	-----------------------	-------------

In *LabelsDE*:

Cl <i>s</i> MyMethod	Meine eigene numerische Methode	Cl <i>s</i>
----------------------	---------------------------------	-------------

In beiden Fällen ist der Key (Name des Eintrags) der Klassenname, der Wert die eigene gewählte Bezeichnung und der Kommentar 'Cl*s*'.

Für andere eigene Erweiterungen bei den Unimodalen Funktionen oder dem Billard gilt genau das analoge. Dort muss je das entsprechende Interface (*IIterator* bzw. *IBilliardball*) implementiert werden. Das kann aber besonders beim Billard aus mathematischer Sicht aufwendiger sein.

Weiterführende Literatur

Aufgeführt sind hier wenige Quellen, bei denen einzelne Kapitel noch in einem elementar zugänglichen Rahmen liegt. Ausgehend von diesen Quellen findet man weitere Literaturhinweise.

- [1] Wolfgang Metzler: Nichtlineare Dynamik und Chaos, Teubner Studienbücher 1998
- [2] Urs Kirchgraber: Mathematik im Chaos, Mathematische Semesterberichte, Springer 1992
- [3] Urs Kirchgraber: Chaotisches Verhalten in einfachen Systemen, Berichte über Mathematik und Unterricht, ETHZ, 1992
- [4] Urs Kirchgraber, Niklaus Sigrist: Feigenbaum Universalität: Beschreibung und Beweisskizze, Berichte über Mathematik und Unterricht, ETHZ, 1995
- [5] Pierre Collet, Jean-Pierre Eckmann: Iterated maps on the Interval as dynamical Systems, Birkhäuser 1986
- [6] Moritz Adelmeyer: Theorem von Sarkovskii, Berichte über Mathematik und Unterricht, ETHZ, 1990
- [7] Serge Tabachnikov: Geometrie und Billard, Springer Spektrum 2013
- [8] Anna-Maria Vocke: Das Poincaré-Birkhoff Fixpunkttheorem und Anwendungen auf Billards, Bachelorarbeit, Mathematisches Institut, Westfälische Wilhelms-Universität Münster, 2014
- [9] Bastian von Harrach: Numerik von Differentialgleichungen, Vorlesungsskript, Goethe-Universität Frankfurt am Main, Institut für Mathematik, 2022
- [10] Robin Santra: Einführung in den Lagrange- und Hamilton-Formalismus, Springer Spektrum 2022
- [11] Stefan Frei: Numerische Mathematik, Vorlesungsskript an der Universität Konstanz, WS 2021,2022
- [12] Hans Jürgen Korsch, Hans-Jörg Jodl, Timo Hartmann: Chaos, Springer Verlag 2008
- [13] Dierk Schleicher, Malte Lackmann, *Hrsg.*: Eine Einladung in die Mathematik, Springer Spektrum 2013
- [14] Reinhold Remmert. Georg Schumacher: Funktionentheorie 1, Springer Verlag 2002