

Benutzerhandbuch «Simulator»

Benutzerhandbuch zum Gebrauch des Programmes «Simulator». Die mathematischen Grundlagen sind im Dokument «Dynamische Systeme» beschrieben.

Version 6.0 - 01.10.2024 - Dr. Hermann Biner

Inhalt

1.	Einführung	2
2.	Hauptmenü.....	2
3.	Menü Billard	4
3.1.	Billard	4
3.2.	C-Diagramm	10
4.	Menü Wachstumsmodelle.....	14
4.1.	Iterationen im reellen Intervall.....	14
4.2.	Untersuchung der Sensitivität	20
4.3.	Histogramme.....	23
4.4.	Zweidimensionale Darstellung.....	24
4.5.	Feigenbaum Diagramm.....	26
5.	Menü komplexe Iteration.....	31
5.1.	Newton Iteration.....	31
5.2.	Julia Menge und Mandelbrot Menge	38
6.	Menü Mechanik.....	46
6.1.	Numerische Methoden	46
6.2.	Pendel	50

1. Einführung

Der «Simulator» ermöglicht die Simulation von einfachen dynamischen Systemen. Die mathematischen Grundlagen dazu und die Konzepte für die Implementierung sind im Dokument «Dynamische Systeme» beschrieben. In diesem Dokument finden sich auch Anregungen für Weiterentwicklungen oder mathematische Übungsaufgaben.

Der Code des Programmes ist in GitHub veröffentlicht und als Open Source zugänglich. Er ist in VB.NET verfasst und ausführlich (auf Englisch) kommentiert. Die Entwicklungsumgebung ist die Community Version von Microsoft Visual Studio 2022. Sie ist gratis verfügbar und einfach zu installieren. Nötig ist mindestens die Version 17.9. Diese setzt auf dem Microsoft Framework 8.0 auf, welches ebenfalls heruntergeladen und installiert werden muss, falls es nicht schon vorhanden ist.

Die aktuelle Version des «Simulator» ist 6.0.0 publiziert am 1.10.2024.

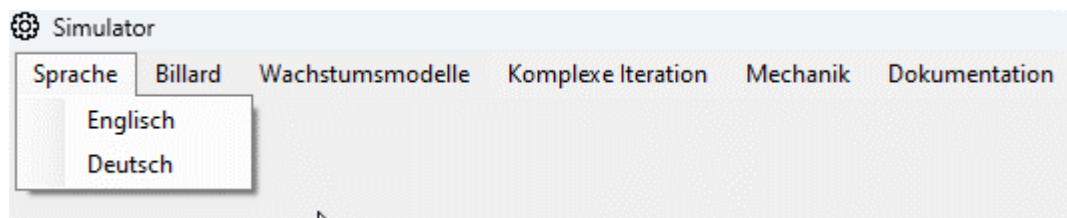
Der GitHub Link ist folgender:

<https://github.com/HermannBiner/Simulator>

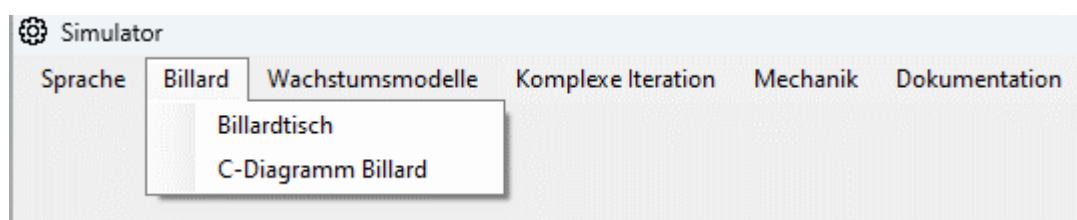
Dieses Handbuch erklärt den Gebrauch des «Simulator» und erläutert Beispiele für dessen Einsatz.

2. Hauptmenü

Unter «Sprache» ist die Sprache wählbar:



Im Menü «Billard» können verschiedene Formen des mathematischen Billards untersucht werden.

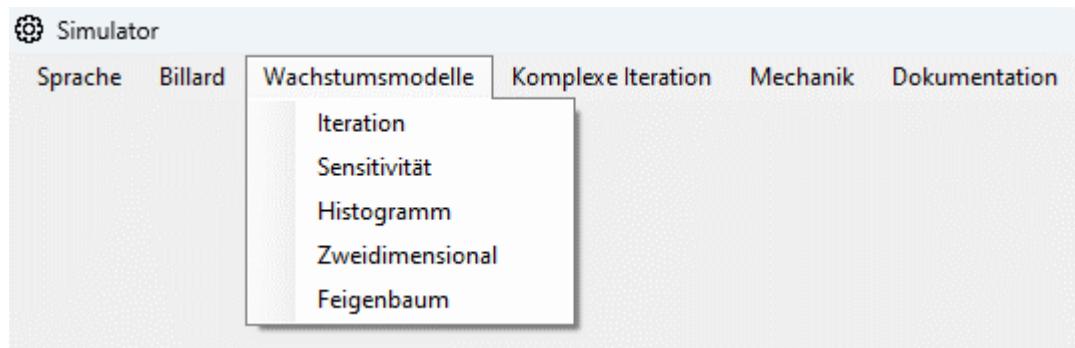


Insbesondere bietet das Untermenü «Billardtisch» folgende Optionen an:

- Elliptisches Billard
- Billard im Stadion
- Ovaless Billard

Die Form des Billardtisches wird überall durch einen Parameter C definiert. Im Falle des elliptischen Billards ist C das Achsenverhältnis der Ellipse. Analog zum Feigenbaum-Diagramm im Falle der quadratischen Funktionen kann man hier ein C-Diagramm erstellen, welches das Verhalten der Iterationsparameter in Abhängigkeit von C darstellt.

Der Bereich «Wachstumsmodelle» ermöglicht verschiedene Experimente, welche auf der Iteration von reellen Funktionen beruhen.

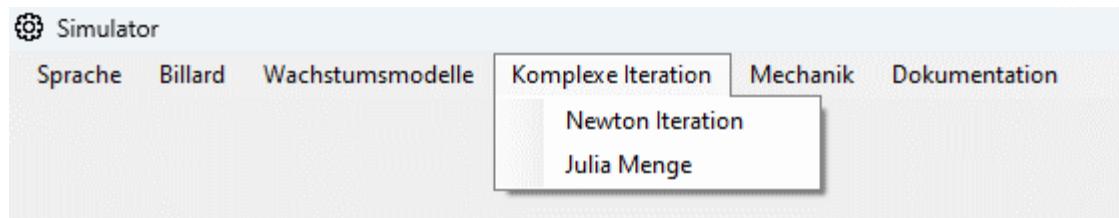


Implementiert sind folgende Funktionen:

- Zeltabbildung
- Logistisches Wachstum
- Iteration der Parabel
- Mandelbrot-Iteration im Reellen (nur für Feigenbaum, d.h. die Untersuchung der Bifurkation bei der Mandelbrotmenge für reelle c)

Die Untermenüs bieten verschiedene Funktionen, unter anderem auch die Untersuchung des chaotischen Falles. Sie werden in diesem Dokument später beschrieben.

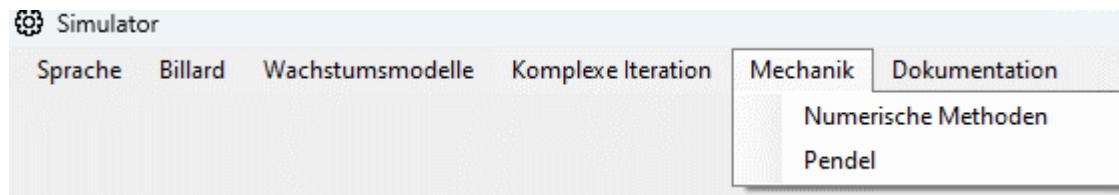
Im Bereich «Komplexe Iteration» werden Iterationen in der komplexen ebene zur Untersuchung angeboten.



Zur Verfügung stehen:

- Newton Iteration (vor allem im Falle der komplexen Einheitswurzeln)
- Erzeugung von Julia Mengen und der dazugehörigen Mandelbrot Menge für die quadratische Funktion und die Potenzfunktion des Grades n

Der Bereich «Mechanik» bietet verschiedene Pendelsysteme an, welche untersucht werden können.



Ferner kann man verschiedene numerische Methoden zum Lösen von gewöhnlichen Differentialgleichungen am Beispiel des Federpendels untersuchen:

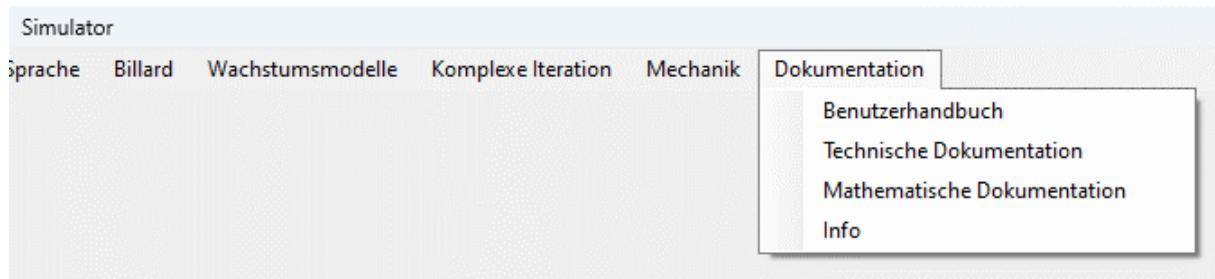
- Euler explizit
- Euler implizit
- Mittelpunktregel

- Runge Kutta

Im Menüpunkt «Pendel» stehen verschiedene Versionen von gekoppelten Pendeln zur Untersuchung bereit:

- Doppelpendel
- Schwingendes Federpendel
- Horizontales Rüttelpendel

Der Menüpunkt «Dokumentation» bietet alle Informationen, welche für den Gebrauch des Programmes nötig sind.



Das umfasst ein Benutzerhandbuch, eine technische Dokumentation und eine mathematische Dokumentation. Ferner sind allgemeine Informationen zum Programm verfügbar.

Alle Dokumente sind je nach gewählter Sprache auf Deutsch oder Englisch verfügbar.

3. Menü Billard

3.1. Billard

Der «Simulator» ermöglicht es, verschiedene Formen des Billards zu untersuchen. Implementiert sind folgende Arten von Billards:

- Elliptisches Billard
- Billard im Stadion
- Ovales Billard

In allen Fällen wird die Form des Billardtisches durch einen Parameter $C > 0$ definiert. Dabei ist:

- C ist das Verhältnis Nebenachse zu Hauptachse beim elliptischen Billard
- C ist das Verhältnis von Kreisdurchmesser zu Rechteckbreite beim Billard im Stadion
- C ist das Verhältnis von Kreisradius (rechter Teil) zu Ellipsenhauptachse (linker Teil)

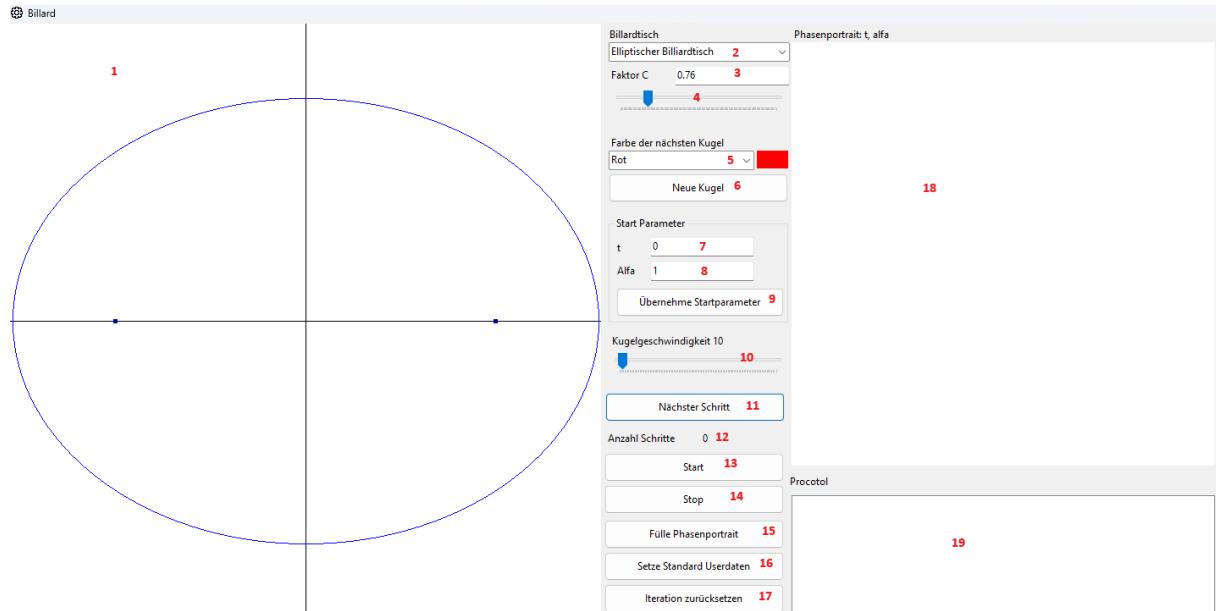
Da der Code öffentlich ist, kann der User weitere Formen des Billards programmieren. Er muss dazu lediglich ein Interface implementieren. Siehe technische Dokumentation oder Kommentare im Code.

Es wird angenommen, dass sich der Billardball reibungsfrei bewegt und bei den Stößen am Rand des Billardtisches keine Energie verloren geht. Es können beliebig viele Billardkugeln auf dem Tisch platziert werden. Diese stossen sich aber nicht gegenseitig. Die dienen dazu, verschiedene Orbits in Abhängigkeit von den Startparametern sichtbar zu machen.

Startparameter sind zuerst immer ein Punkt auf den Rand des Billardtisches, welcher gemäss mathematischer Dokumentation durch einen Parameter t beschrieben wird. Als zweiter Parameter

dient der Reflexionswinkel α (im Bogenmass) zwischen Kugelbahn und Tangente im Stoßpunkt des Billardtisches.

Bei allen Arten des Billards ist das Vorgehen zu deren Untersuchung dasselbe. Im Menü «Mechanik – Billard» wird folgendes Fenster geöffnet:



Fenster zur Untersuchung der verschiedenen Billards

1

Das ist die Zeichenfläche für den Billardtisch und die Kugelbahnen.

2

Hier wird die Art des Billardtisches gewählt. Siehe Liste am Anfang.

3

Hier kann die Form des Billardtisches durch einen Parameter C definiert werden gemäss früherer Erläuterung. Dieses Verhältnis muss eine positive Zahl sein. Für den Wert C = 1 erhält man im Falle des elliptischen oder ovalen Billards einen kreisförmigen Billardtisch.

4

Mit dieser Scrollbar kann man den Parameter C und damit die Form des Billardtisches einstellen. Der Billardtisch wird dann direkt gezeichnet.

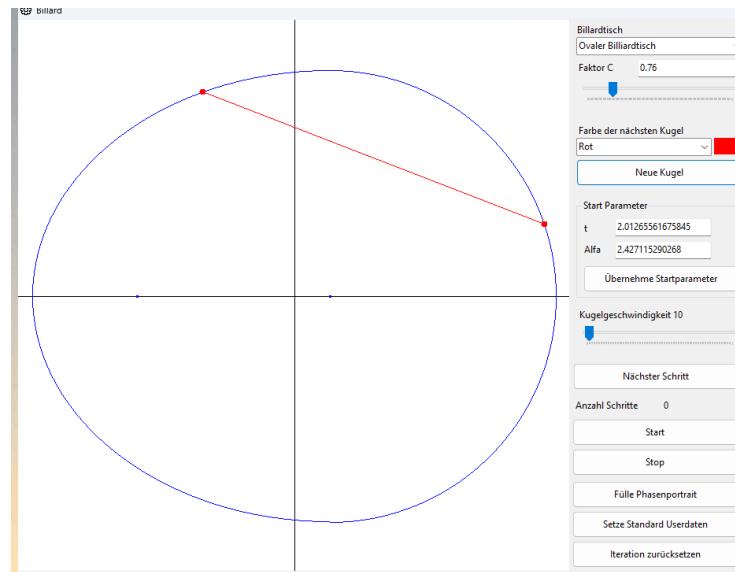
5

Die Kugel ist so programmiert, dass beliebig viele Instanzen (also Kugeln) generiert werden können. Damit diese unterschieden werden, kann je eine von fünf Farben für die nächste generierte Kugel festgelegt werden.

6

Diese Schaltfläche generiert eine neue Kugel. Diese wird standardmäßig platziert, je nach Art des Billardtisches. Anschliessend die Kugel mit gedrückter linker Maustaste an ihrem Startpunkt platziert werden. Wird die Maustaste losgelassen, gilt der Startpunkt t als gesetzt. Wird die Maustaste zum

zweiten Mal gedrückt, kann jetzt die Richtung des ersten Bahnabschnittes gesetzt werden. Wird die Maus wieder losgelassen, gilt auch der erste Bahnwinkel α als gesetzt.



Ovaler Billardtisch mit gesetzter Kugel inklusive erstem Bahnabschnitt

Wenn die Kugel und ihr erster Bahnabschnitt auf diese Art manuell platziert wird, dann werden rechts im Bereich «Startparameter» die Parameter t (Position des Startpunktes auf dem Rand) und α (erster Winkel zwischen Bahnabschnitt und Tangente am Billardtisch) angezeigt. In gewissen Fällen ist es auch nützlich, diese Parameter manuell einzugeben und sie dann als Startparameter zu übernehmen.

7, 8

Hier werden die Startparameter t, α angezeigt. Diese Werte können auch manuell gesetzt werden. Das ist nötig, wenn man durch ein Näherungsverfahren bestimmte Bahnen erreichen will. Nach einer solchen manuellen Definition der Startparameter können diese durch die Taste

9

übernommen werden. Der Ball wird dann entsprechend platziert. Da die Parameter t modulo 2π oder modulo dem Umfang U des Billardtisches und α modulo π gerechnet werden, sind alle reellen Startparameter erlaubt. Damit ist $t \in [0, 2\pi[$ oder $t \in [0, U[$ und $\alpha \in]0, \pi[$.

10

Hier kann man die Geschwindigkeit aller Kugeln für die nächsten Stöße gleichzeitig regeln.

11

Hier wird ein einzelner Stoß durchgeführt, und zwar von allen beteiligten Kugeln.

12

Hier wird die Anzahl der Schritte angezeigt.

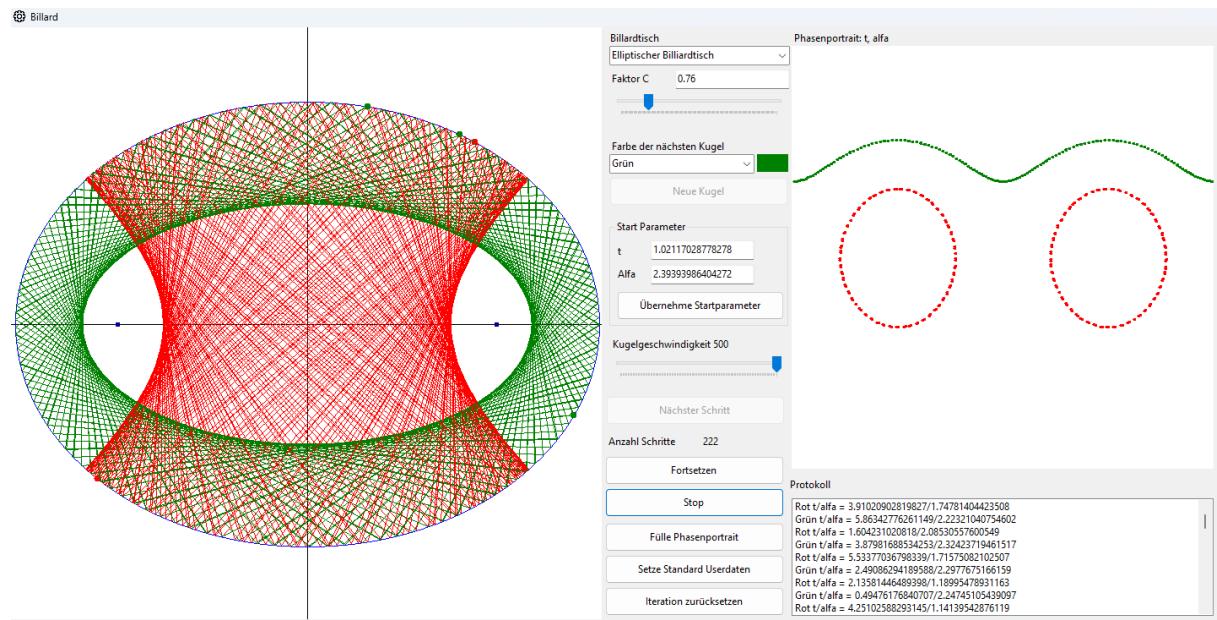
13

Hier führen alle Kugeln Stöße durch bis der Vorgang durch die Stoptaste

14

gestoppt wird.

Beispiel

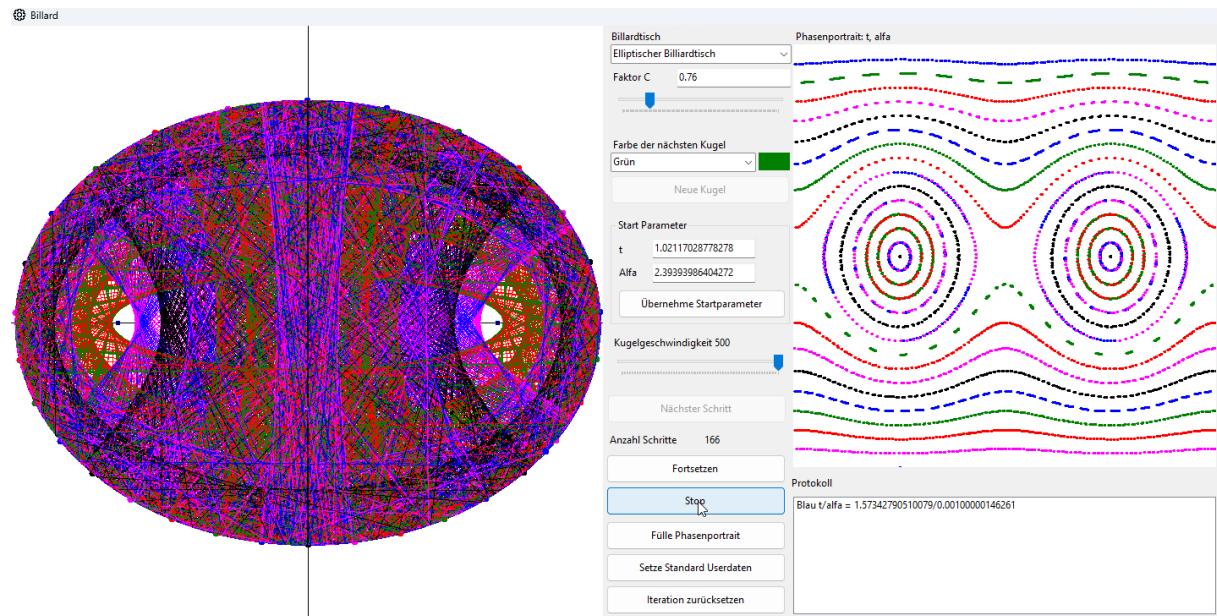


Elliptisches Billard mit zwei Kugeln

Im obigen Beispiel wurden zwei Kugeln platziert. Die rote Bahn läuft zwischen den Brennpunkten der Ellipse und die grüne ausserhalb davon. In der mathematischen Dokumentation werden die auftretenden Kaustiken hergeleitet.

15

Hier werden viele Billardkugeln erzeugt und über den Tisch verteilt, damit anschliessend durch «Start» das Phasenportrait erzeugt werden kann.



Skizze der Ballbewegungen im Phasenraum

16

Durch diese Taste wird die gesamte Iteration zurückgesetzt und die Startparameter t und Alfa auf die Standardvorgaben gesetzt.

17

Durch diese Taste wird nur die Iteration zurückgesetzt, aber die vom Benutzer gesetzten Startparameter werden beibehalten.

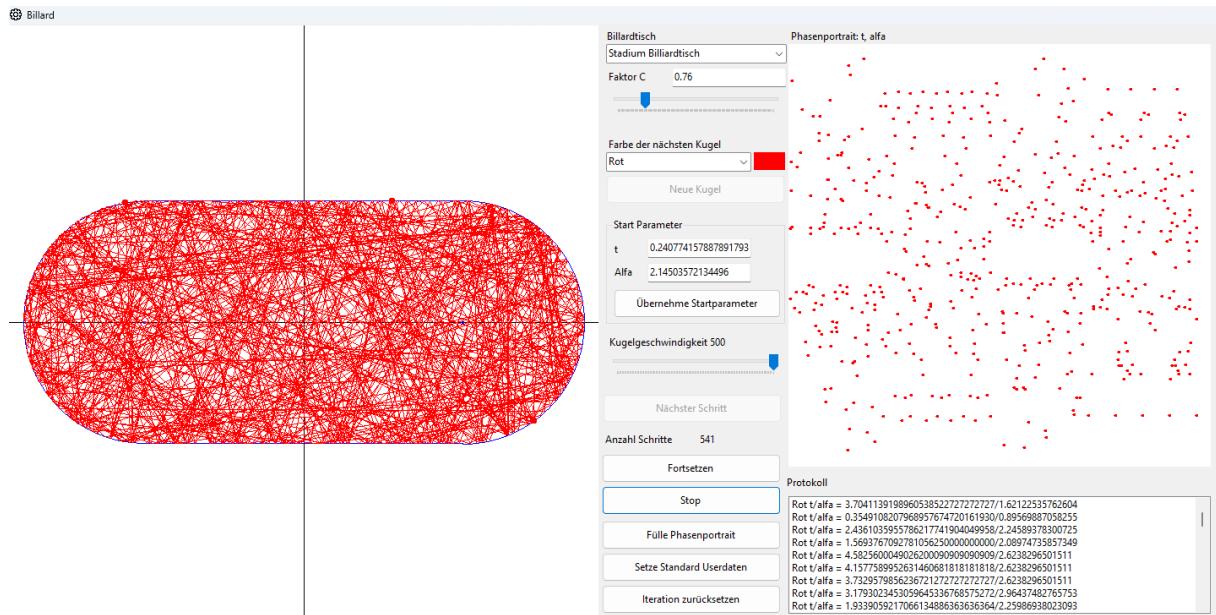
18

Hier wird das «Phasenportrait» der Bewegung skizziert: In horizontaler Richtung wird der Parameter t und in vertikaler Richtung der Parameter α eingetragen. Man sieht im obigen Beispiel, dass im grünen Fall t alle mögliche Werte durchläuft, während α auf einen gewissen Bereich eingeschränkt ist. Im roten Fall sind sowohl t wie auch α eingeschränkt.

19

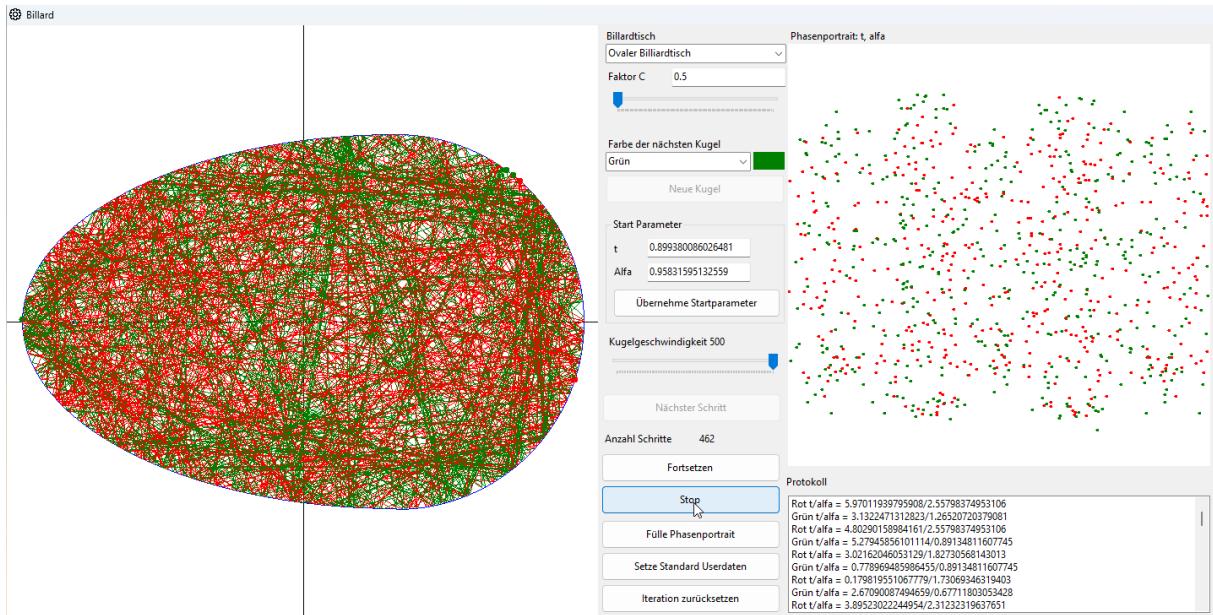
Hier werden die Parameterpaare (t_n, α_n) aufgelistet, und zwar unterschieden nach der Kugelfarbe.

Beispiele



Eine Kugel im stadionförmigen Billard

Rechts im Phasenportrait erscheint keine Struktur mehr. Die Parameterpaare scheinen zufällig verstreut.

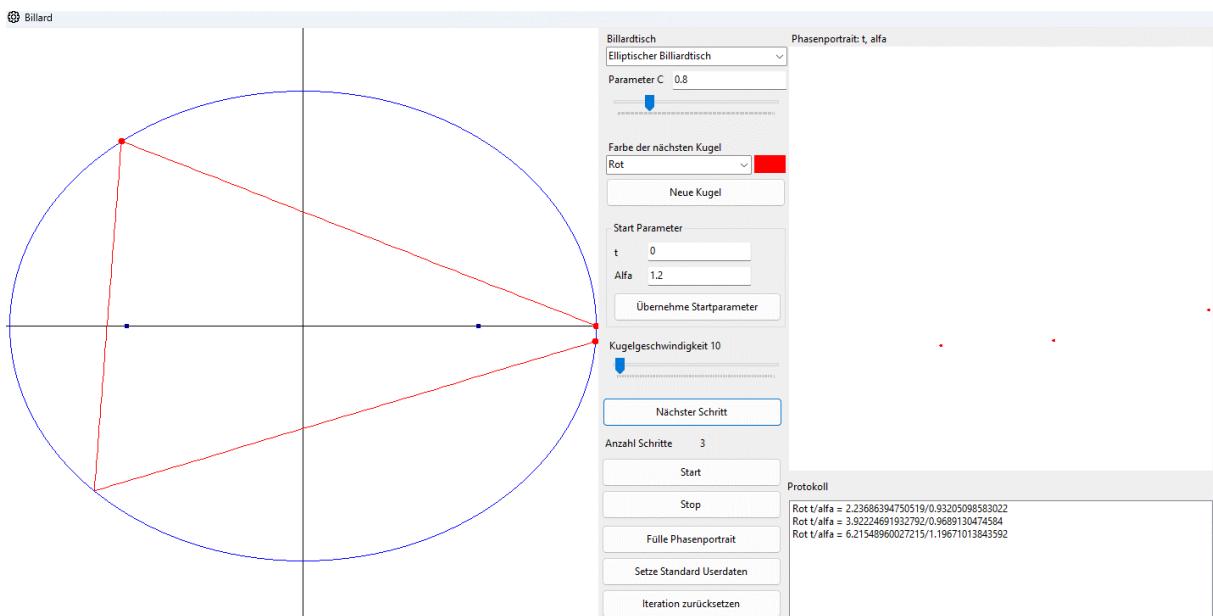


Zwei Kugeln im ovalen Billard

Auch hier erscheint keine Struktur mehr im Phasenportrait.

Ein weiteres Beispiel zeigt die Möglichkeit, Zyklen durch Intervallschachtelung näherungsweise zu bestimmen.

Angenommen, wir suchen einen 3-periodischen Zyklus im Falle des elliptischen Billards für den Parameter $C = 0.8$. Wir wählen als Startpunkt für den Zyklus jeweils den Parameter $t = 0$, also den Startpunkt $(a, 0)$. Dann sehen wir mit etwas experimentieren, dass der Startwinkel $\alpha = 1.2$ eine erste sehr grobe Näherung liefert. Wir tragen diese Werte manuell als Startparameter ein, übernehmen die Startparameter (dadurch wird die Kugel platziert) und führen die ersten drei Stöße durch:



Suche nach dem 3-Zyklus: Der Wert $\alpha = 1.2$ ist etwas zu klein

Das sieht man hier noch optisch. Es ist aber besser, dass Protokoll rechts zu kontrollieren und man sieht, dass $t = 6.215489 \dots$ nicht ganz 2π beträgt.

Wie man ebenso feststellt, ist $\alpha_2 = 1.25$ etwas zu gross. Wir geben diese Werte jeweils manuell im Parameterbereich ein und lassen die Kugel die Parameterwerte übernehmen. Wir kontrollieren den Parameter t nach jeweils drei Stößen. Wenn $t > 0$ ist, dann war α etwas zu gross. Wenn t knapp unter 2π liegt, dann war α etwas zu klein. Je nachdem korrigieren wir das durch Halbieren des Intervalls aus. Das ergibt folgende Intervallsschachtelung für den gesuchten Startwert α :

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

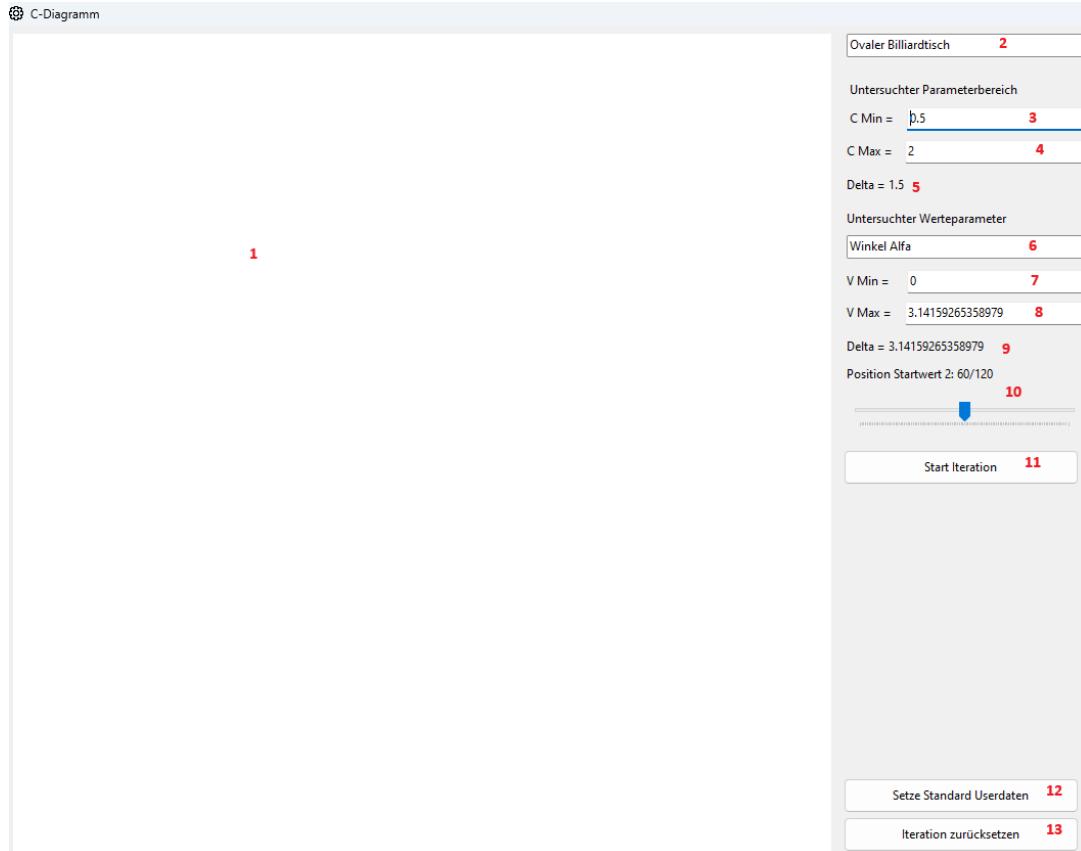
Der gesuchte 3-er Zyklus startet im Bereich $t = 0, \alpha \in [1.2226, 1.2235]$

Das Verfahren konvergiert sehr langsam. Auf Grund von Stetigkeitsargumenten kann man aber mindestens auf die Existenz eines 3-er Zyklus schliessen. Siehe mathematische Dokumentation.

3.2. C-Diagramm

Im Falle der quadratischen Funktionen zeigt das Feigenbaum-Diagramm das Verhalten der Iteration in Abhängigkeit des Parameters a . Beim Billard spielt diese Rolle der Parameter C .

Für die Darstellung des C-Diagrammes wird folgendes Fenster geöffnet:



Fenster zum Untersuchen des C-Diagrammes

1

Hier wird die Grafik angezeigt.

2

Hier wird die Form des Billardtisches ausgewählt.

3, 4, 5

Hier kann festgelegt werden, welches Intervall von C untersucht wird. Es gilt: $C \in [C_{min}, C_{max}]$. Die Breite des untersuchten Intervalls wird in 5 angezeigt.

6

Die Bewegung der Kugel wird durch zwei Werteparameter beschrieben. Hier wird der Werteparameter ausgewählt, dessen Abhängigkeit von C dargestellt werden soll.

7, 8, 9

Für den ausgewählten Werteparameter wird hier festgelegt, welches Intervall näher untersucht werden soll. Der Werteparameter ist dann im Intervall $[V_{min}, V_{max}]$. Die Breite dieses Intervalls wird in 9 angezeigt.

10

Der erste Werteparameter beim Billard beschreibt die Position des Stosspunktes auf dem Rand des Billardtisches. Der zweite Werteparameter den Reflexionswinkel beim Stoss.

Der Werteparameter, welcher in 6 *nicht* ausgewählt ist, das heisst, der nicht genauer untersucht wird, wird standardmässig auf 1/3 seines Wertebereichs gesetzt. Also zum Beispiel $t = 2\pi/3$ im Falle des elliptischen Billards.

Für den Werteparameter, der untersucht werden soll, kann die Ausgangsposition zwischen 1/12 bis 11/12 seines Wertebereichs variiert werden. Also zum Beispiel $\alpha = 7\pi/12$ bei der Einstellung, welche bei 10 im obigen Fenster sichtbar ist.

11

Hier wird die Iteration gestartet oder gestoppt.

12

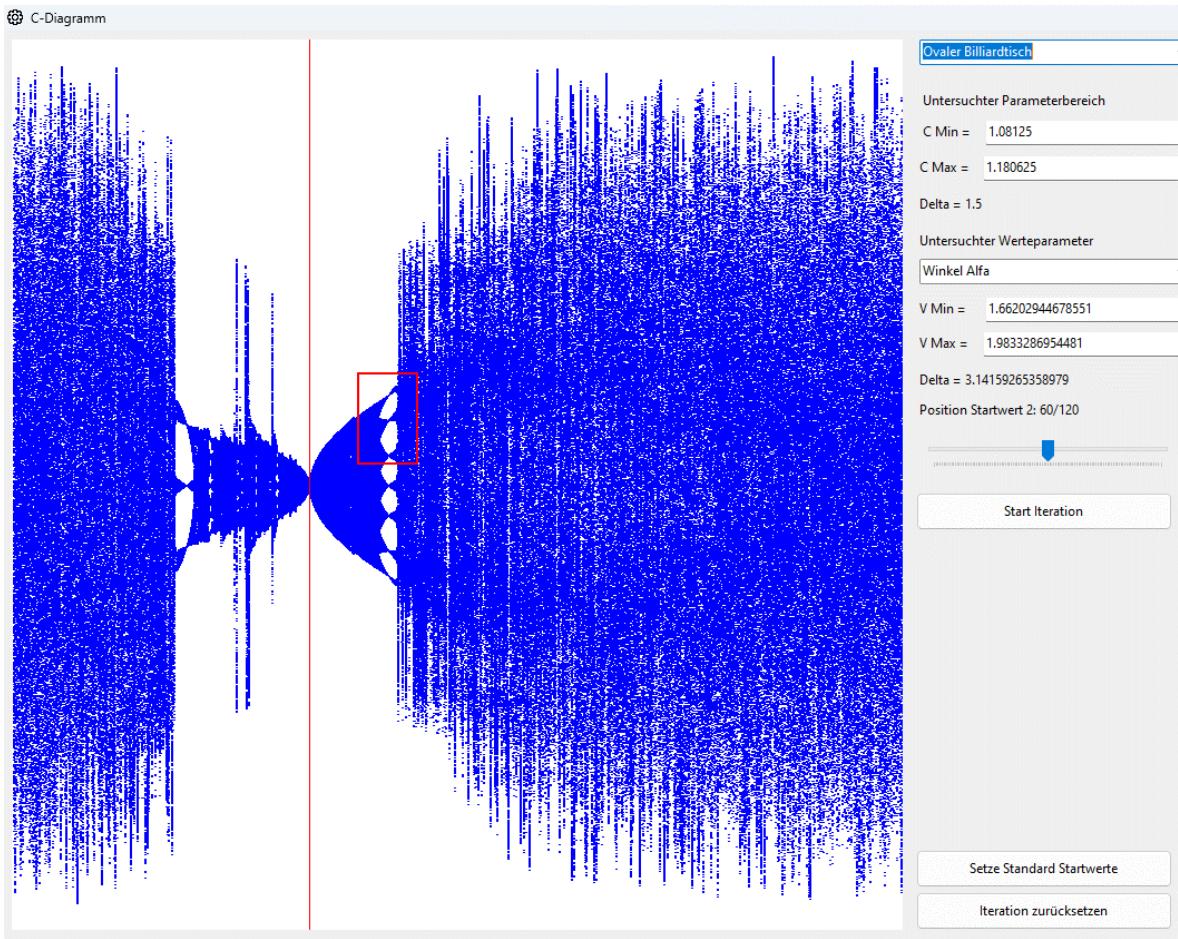
Hier wird die Iteration zurückgesetzt und die Parameterintervalle auf den Standard gesetzt.

13

Hier wird nur die Iteration zurückgesetzt.

Auswahl von kleinen Ausschnitten im Diagramm

Wir starten das C-Diagramm für das ovale Billard mit den vorgegebenen Standardparametern.



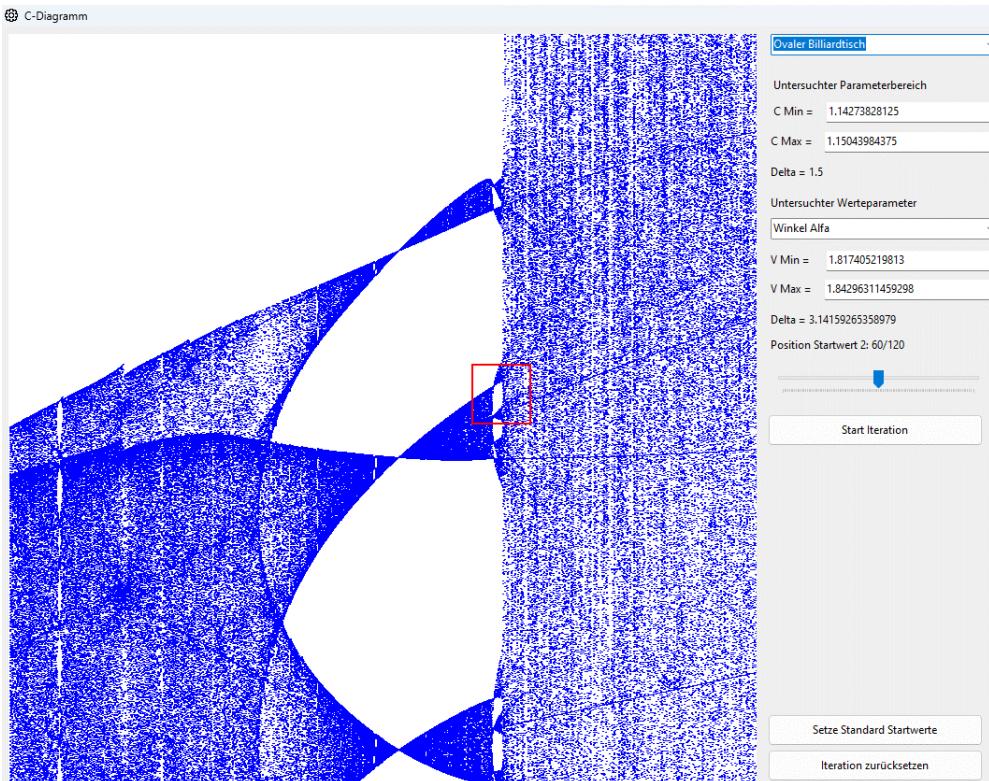
Untersuchung von α beim ovalen Billard mit Benutzerauswahl – siehe rotes Rechteck

Oben wurde das ovale Billard für den Parameterbereich $C \in [0.5, 2]$ untersucht, und zwar der Reflexionswinkel α . Die rote senkrechte Linie markiert den Parameterwert $C = 1$. Für diesen Wert ist der Billardtisch ein Kreis und α ist konstant, d.h. man hat einen Punkt im Diagramm.

Der Benutzer hat nun die Möglichkeit, einen Ausschnitt, den er genauer untersuchen will, mit Hilfe eines Auswahl-Rechteckes auszuwählen. Das geschieht mit gedrückter linker Maustaste. Dabei wird rechts in den Feldern 3-5 der Parameterbereich und in den Feldern 6-8 der Wertebereich automatisch entsprechend nachgefahren.

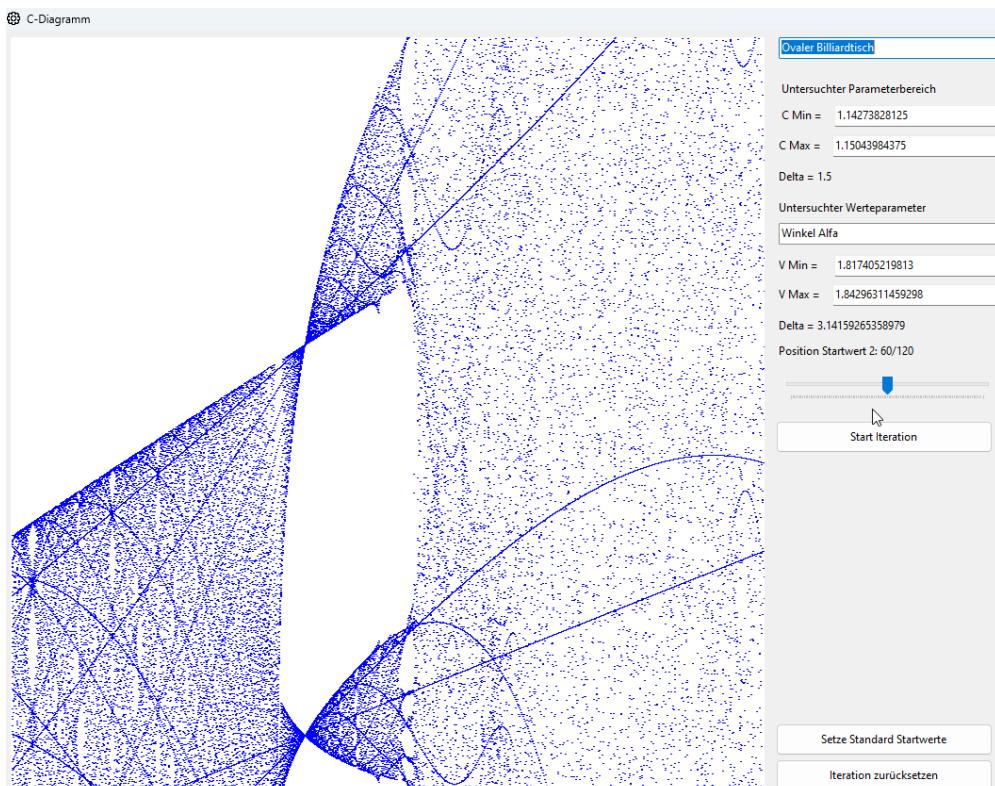
Im oberen Bild hat der User mit gedrückter linker Maustaste den rot markierten Ausschnitt gewählt. Dabei werden rechts die Intervalle $[C_{min}, C_{max}]$ und $[V_{min}, V_{max}]$ nachgefahren, wie beim Feigenbaum-Diagramm.

Wird nun die Iteration nochmals gestartet, erhält man folgendes Bild:



Ausschnitt aus dem C-Diagramm für das ovale Billard

Hier hat der Benutzer einen weiteren Ausschnitt zur Vergrösserung ausgewählt. Wenn man die Iteration mit diesem Ausschnitt startet, erhält man:



Ein weiterer Ausschnitt aus dem Diagramm

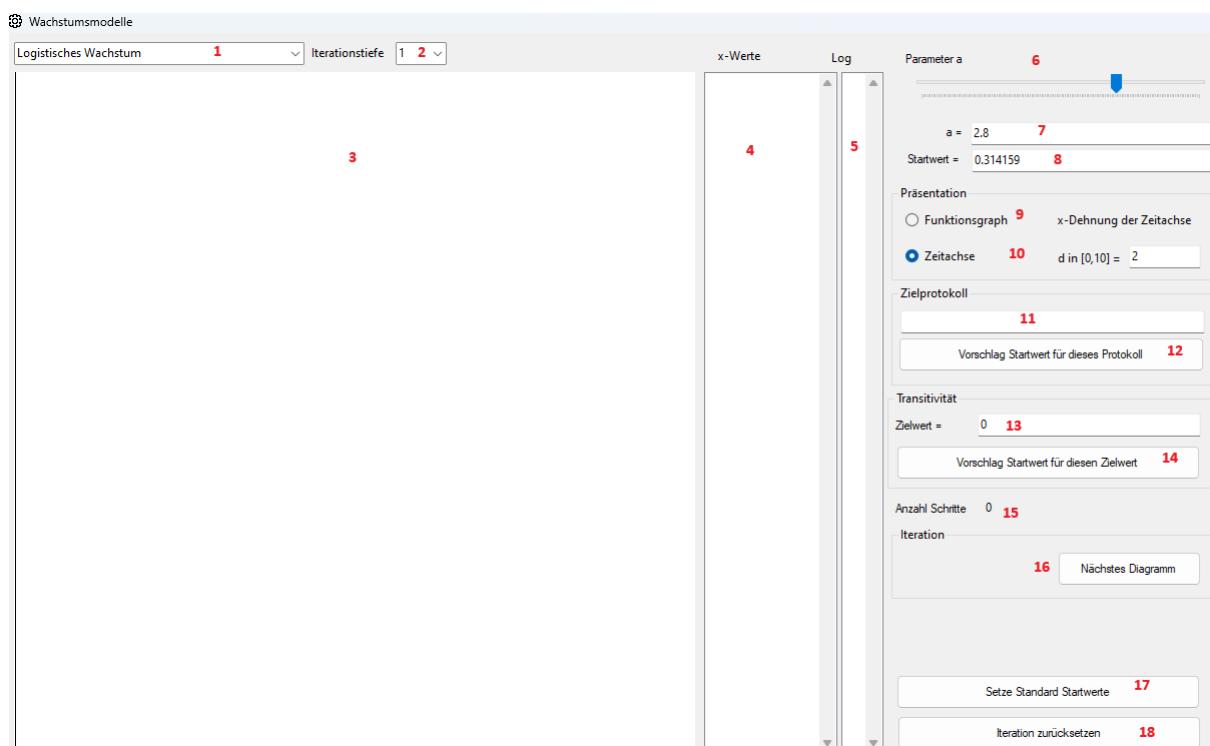
Bemerkung für weitere Experimente

Meist ist die Untersuchung des Reflexionswinkels α aussagekräftiger als der Parameter t , welcher den Stoßpunkt festlegt. Beim Kreis oder bei einem Orbit ausserhalb der Ellipsenbrennpunkte liegt der Orbit von t dicht im zulässigen Intervall, ausser man hat gerade einen periodischen Orbit erwischt. Hingegen ist α beim Kreis konstant oder oszilliert bei der Ellipse in einem Teilintervall von $[0, \pi]$. Bevor man mit einer Untersuchung beginnt, kann man zudem mit der Position des Startwertes experimentieren, um interessante Bilder zu erhalten. Deren vertiefte mathematische Interpretation dürfte aber schwierig sein.

4. Menü Wachstumsmodelle

4.1. Iterationen im reellen Intervall

Über den Menüpunkt «Wachstumsmodelle - Iteration» wird folgendes Fenster geöffnet:



Fenster für die Untersuchung von Iterationen im «Simulator»

1

Links oben wird die Art der Iteration ausgewählt. Zur Verfügung stehen:

- Zeitabbildung
- Logistisches Wachstum
- Iteration der Parabel

Die Iterationen sind wie folgt definiert:

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

$$\text{Logistisches Wachstum: } f: [0,1] \rightarrow [0,1], x \mapsto ax(1-x), a \in]0,4]$$

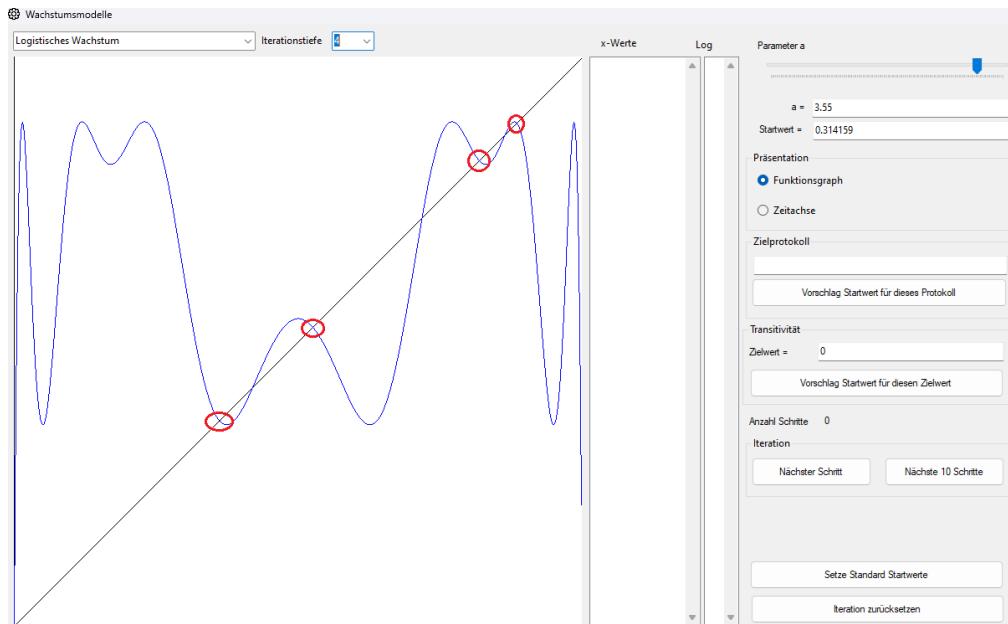
$$\text{Iteration der Parabel: } f: [-1,1] \rightarrow [-1,1], x \mapsto 1 - ax^2, a \in]0,2]$$

Der User kann weitere Arten von Iterationen programmieren. Dazu muss er lediglich ein Interface implementieren. Siehe technische Dokumentation oder Kommentare im Code.

2

Hier wird die Iterationstiefe festgelegt. Sie bestimmt, wie oft die Iterationsfunktion f pro Iterationsschritt wiederholt wird. z.B. gilt für Iterationstiefe = 1: $x_{n+1} = f(x_n)$. Bei einer Iterationstiefe = 5 gilt: $x_{n+1} = f^5(x_n)$. Das ist für die Iteration selbst nicht unbedingt wichtig, wohl aber, wenn Zyklen höherer Ordnung anhand des Funktionsgraphen untersucht werden.

Zum Beispiel hat man bei der logistischen Abbildung für den Parameterwert $a = 3.55$ einen attraktiven 4-Zyklus.



Attraktiver 4-Zyklus

Um das sichtbar zu machen, wählt man «Logistisches Wachstum» als Iteration. Als Parameterwert $a = 3.55$ und als Iterationstiefe 4. Das heisst, man untersucht die vierfach iterierte Funktion f^4 . Lässt man ihren Graphen zeichnen, zeigt sich, dass sie die 45° Gerade an vier Stellen schneidet, an denen der Multiplikator (die Steigung der Tangente) kleiner 1 ist und dass somit dieser Zyklus attraktiv ist. Man sieht auch vier weitere Schnittstellen mit der 45° Geraden, welche zu einem repulsiven Zyklus gehören, denn die Tangentensteigung ist hier grösser als 1.

Eine detaillierte Beschreibung findet man in der mathematischen Dokumentation.

3

Das ist der Bereich für alle grafische Darstellungen.

4

Hier werden die Folgenglieder (x_n) aufgelistet, welche durch die Iteration erzeugt werden.

5

In dieser Spalte wird das Protokoll erzeugt, und zwar zu jedem Iterationswert x_n in der linken Liste. Im Falle der Zeitabbildung und des logistischen Wachstums ist das Protokoll $p(x_n)$ definiert als:

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

Im Falle der Iteration der Parabel ist es definiert als:

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

6

Der Parameterwert a für die Iterationsfunktion kann durch eine Scrollbar definiert werden. Wird der Parameterwert verändert, dann wird bei der Darstellungsoption «Funktionsgraph» der Graph der Iterationsfunktion automatisch gezeichnet.

7

Hier wird der Parameterwert a angezeigt oder kann hier manuell eingegeben werden.

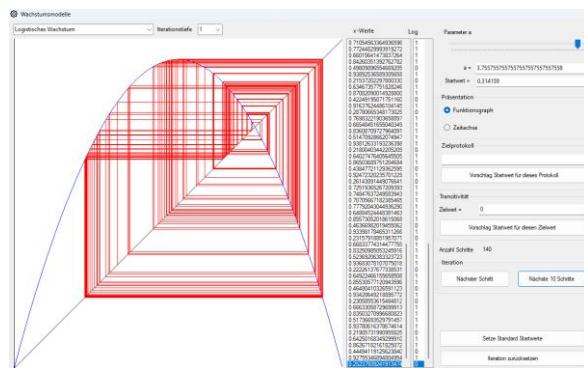
- Zeltabbildung: a muss im Bereich $]0, 2]$ sein
- Logistisches Wachstum: a muss im Bereich $]0, 4]$ sein
- Parabel: a muss im Bereich $]0, 2]$ sein

8

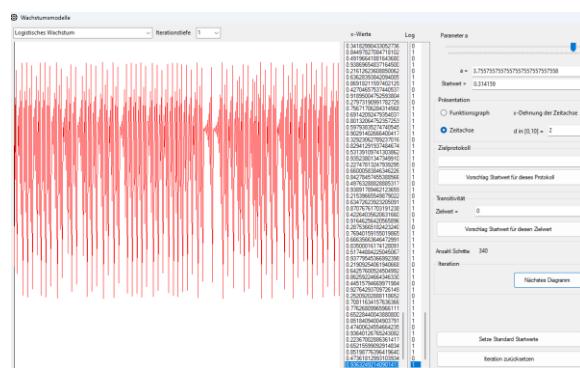
In diesem Feld wird der Startwert festgelegt. Er kann manuell vorgegeben werden oder wird durch das Programm vorgeschlagen. Beim logistischen Wachstum und der Zeltabbildung muss er im Intervall $]0, 1]$ liegen. Bei der Parabel liegt er im Intervall $[-1, 1]$.

9, 10

Um die Iteration zu untersuchen, hat man zwei Darstellungsmöglichkeiten:



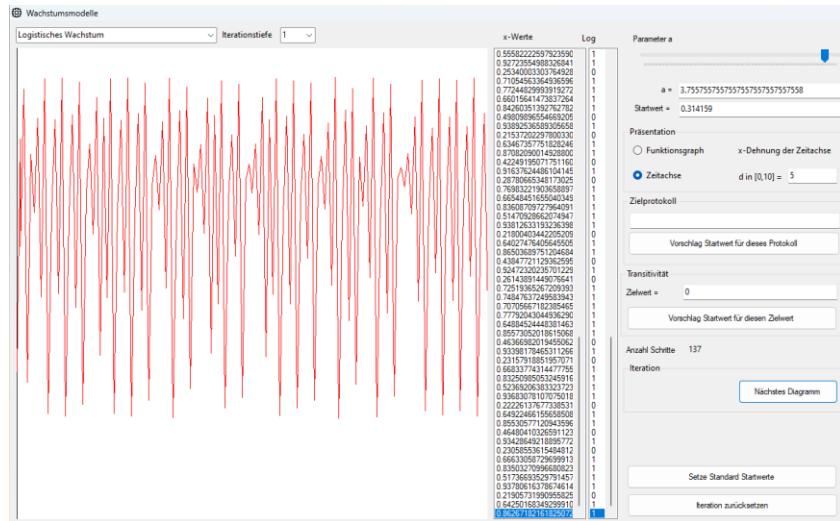
Darstellung im Funktionsgraph



Dieselbe Iteration auf der «Zeitachse»

Bei der Darstellung im Funktionsgraph pendelt man zwischen der 45° Geraden und dem Funktionsgraph hin und her. Die «Zeitachse» zeigt in horizontaler Richtung die Anzahl der Iterationsschritte und vertikal den jeweiligen Iterationswert.

Wenn die Darstellung «Zeitachse» gewählt wird, kann es sein, dass die Iteration nicht mehr auf der Zeitachse erkennbar ist, weil die Sprünge zu eng sind. Dann kann die Zeitachse, welche im Koordinatensystem auf der x-Achse dargestellt wird, gedehnt werden. Möglich ist ein Wert zwischen 1 und 10.



Dieselbe Iteration wie oben, aber mit der Dehnung 5 statt 2

11

Hier kann im chaotischen Fall ein beliebiges Protokoll bestehend aus «0» und «1» vorgegeben werden. Wegen der Rechengenauigkeit des Computers sind Protokolle der Länge bis mindestens 50 möglich, je nach Art der Iteration. Der chaotische Fall liegt u.a. für folgende Parameterwerte vor:

- Zeltabbildung $a = 2$
- Logistisches Wachstum $a = 4$
- Iteration der Parabel $a = 2$

Wenn das Verhalten nicht chaotisch ist, kann ein vorgegebenes Protokoll nicht erzeugt werden.

Beispiel

Wir wählen das Logistische Wachstum mit Parameter $a = 4$, also den chaotischen Fall.

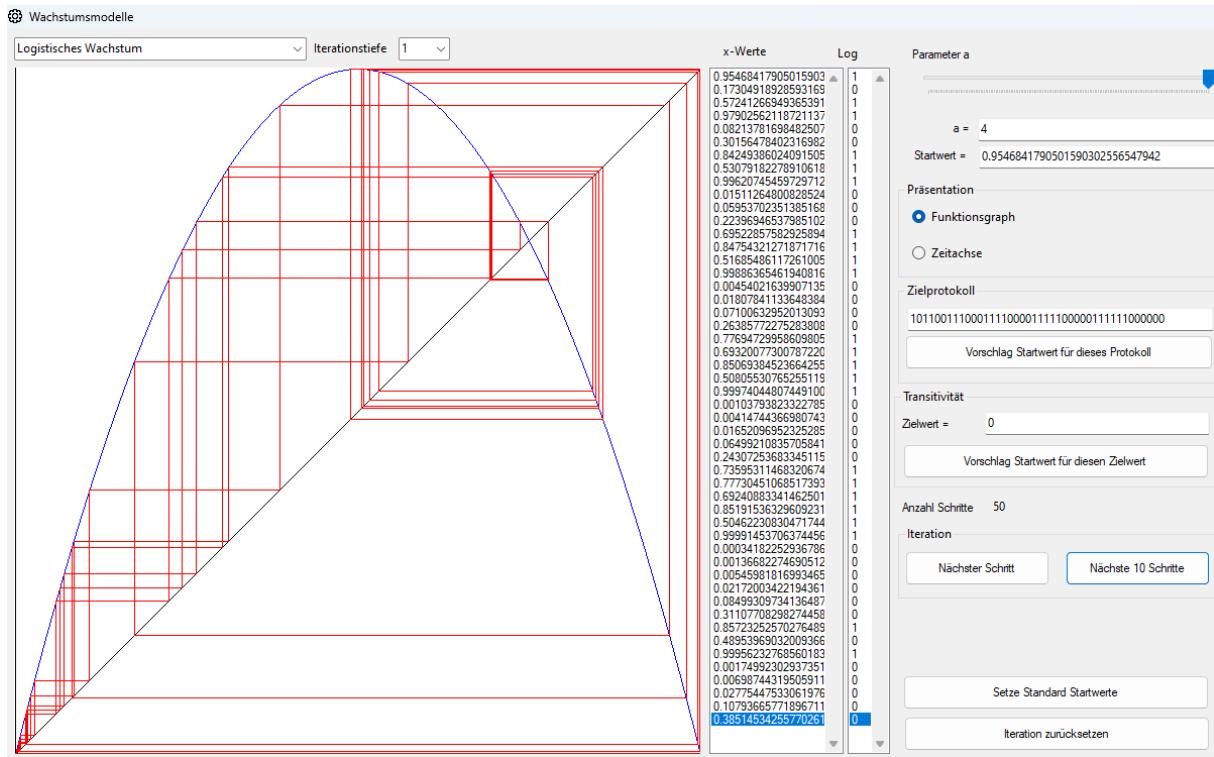
Nun geben wir als Zielprotokoll eine 42-stellige 0-1 Folge ein:

101100111000111100001111100000111111000000

12

Anschliessend drücken wir die Taste: «Vorschlag Startwert für dieses Protokoll». Der «Simulator» schlägt dann als Startwert vor: $x_1 = 0.9546841790501590302556547942$.

Nun drücken wir einige Male die Taste «Nächste 10 Schritte». Im Log für das Protokoll können wir feststellen, dass das vorgegebene Protokoll tatsächlich erzeugt wurde.



Erzeugung eines vorgegebenen Protokolls

Das heisst, dass diese Iteration chaotisch im Sinne des Münzwurfs ist: Zu jedem beliebigen vorgegebenen Protokoll gibt es einen Startwert, welcher dieses Protokoll als Resultat liefert.

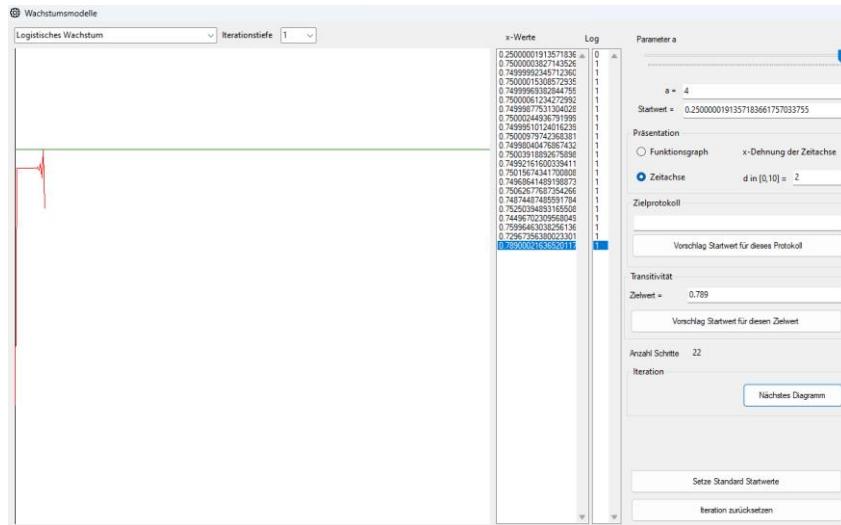
13

Wenn die Iteration chaotisch ist, kann auch die Transitivitt untersucht werden. Dazu wird im Feld 8 ein Startwert und im Feld 13 ein Zielwert vorgegeben. Dann sagt die Theorie, dass es im chaotischen Fall einen leicht genderten Startwert gibt, so dass die Iteration dem Zielwert beliebig nahekommt.

Wir wählen als Startwert z.B. 0.25. Als Zielwert geben wir im Feld 13 den Wert 0.789 vor. Dann betätigen wir die Taste

14

Diese berechnet den leicht abgeänderten Startwert 0.2500000191357183661757033755. Anschliessend wählen wir als Darstellung «Zeitachse» und betätigen die Taste «Nächstes Diagramm». Dann läuft die Iteration wenige Schritte und stoppt dann, wenn sie dem Zielwert nahekommt. Das ist hier beim Iterationswert 0.789000216... der Fall – siehe unten. Der Zielwert wird durch die grüne horizontale Linie markiert.



Nach wenigen Schritten kommt man dem Zielwert nahe.

15

Hier wird die Anzahl der Iterationsschritte angezeigt.

16

Die Taste «Nächster Schritt» ist nur sichtbar im Falle der Darstellung als Funktionsgraph. Sie führt den nächsten Iterationsschritt aus. Bei der Darstellung «Funktionsgraph» werden bei der Taste «Nächste 10 Schritte» jeweils die nächsten 10 Iterationsschritte ausgeführt. Bei der Darstellung «Zeitachse» heisst die Taste «Nächstes Diagramm» und es werden so viele Iterationsschritte ausgeführt, bis das Diagramm in der ganzen Breite gefüllt ist. Wird die Taste nochmals gedrückt, wird die Iteration fortgesetzt und ein neues Diagramm erstellt.

17

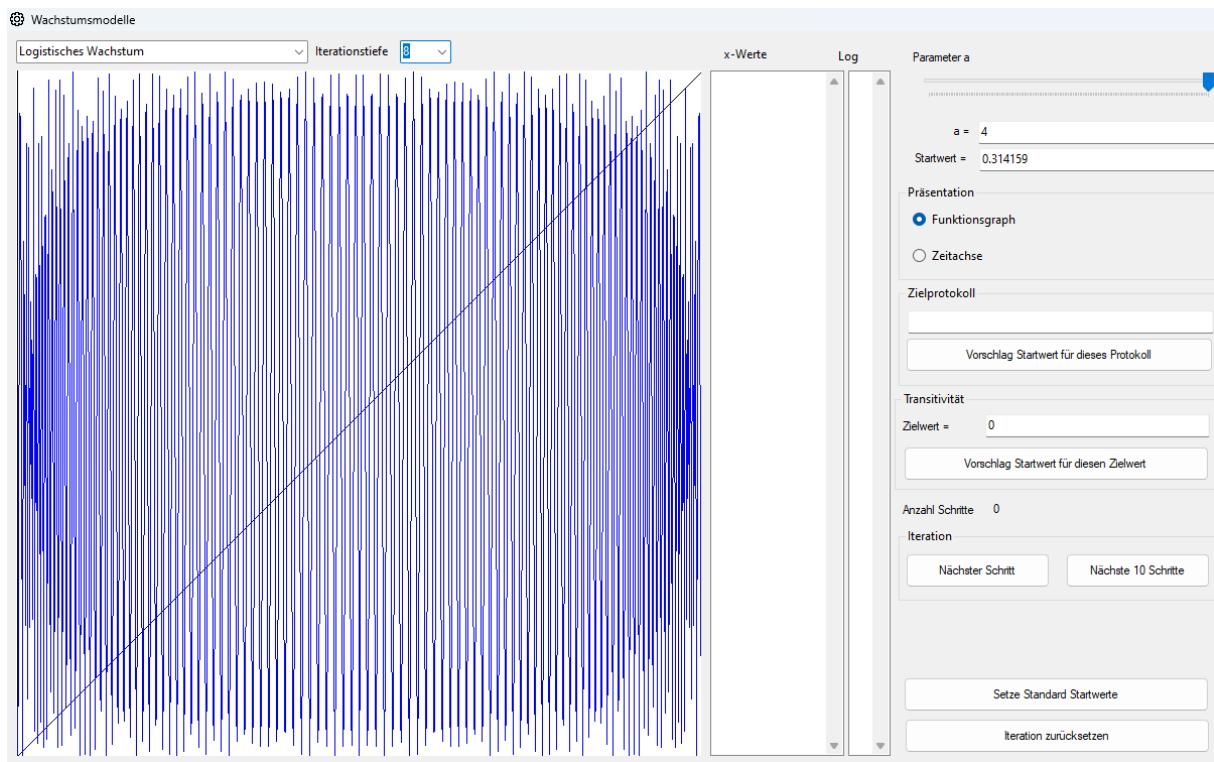
Hier wird die Iteration zurückgesetzt und die Startparameter werden auf den Standard gesetzt.

18

Hier wird lediglich die Iteration zurückgesetzt.

In der mathematischen Dokumentation wird der chaotische Fall für $a = 4$ beschrieben. Ein dynamisches System ist chaotisch im Sinne von Devaney, wenn es die Eigenschaft der Transitivität und Sensitivität hat und wenn zudem die Menge der repulsiven Zyklen dicht im Iterationsintervall sind. Die Transitivität haben wir hier gezeigt.

Für die Dictheit der repulsiven Zyklen kann man den Graphen der iterierten Funktionen betrachten. Zyklen sind Schnittpunkte von deren Graphen mit der 45° Geraden. Diese sind allesamt repulsiv und liegen immer enger beieinander je höher die Periode des Zyklus ist.

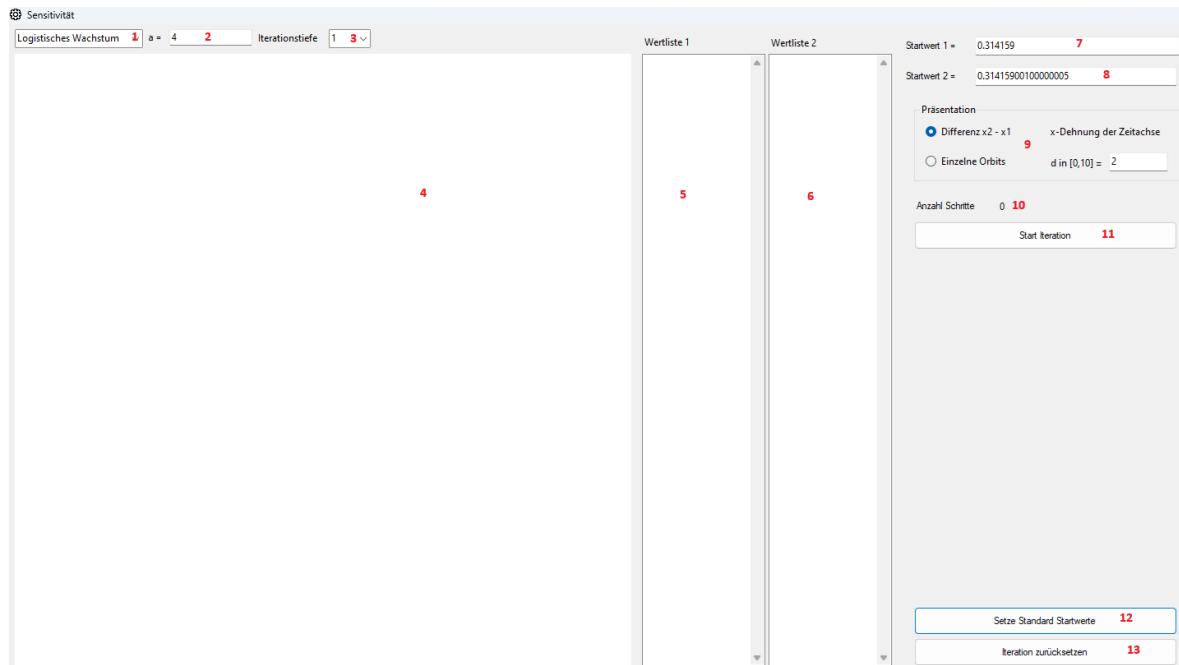


Graph der Iterierten f^8

Die Sensitivität kann im folgenden Abschnitt gezeigt werden.

4.2. Untersuchung der Sensitivität

Unter dem Menüpunkt «Wachstumsmodelle - Sensitivität» wird folgendes Fenster geöffnet:



Fenster zur Untersuchung der Sensitivität mit dem «Simulator»

1

Links oben wird die Art der Iteration ausgewählt. Zur Verfügung stehen:

- Zeltabbildung
- Logistisches Wachstum
- Parabel

2

Der Parameterwert für jede Iteration kann hier eingegeben werden.

- Zeltabbildung: a muss im Bereich $]0,2]$ sein
- Logistisches Wachstum: a muss im Bereich $]0,4]$ sein
- Parabel: a muss im Bereich $]0,2]$ sein

3

Hier wird die Iterationstiefe festgelegt. Sie bestimmt, wie oft die Iterationsfunktion f pro Iterationsschritt wiederholt wird.

4

Das ist der Bereich für grafische Darstellungen.

5

Hier werden die Folgenglieder der Iteration ausgehend vom ersten Startwert aufgelistet.

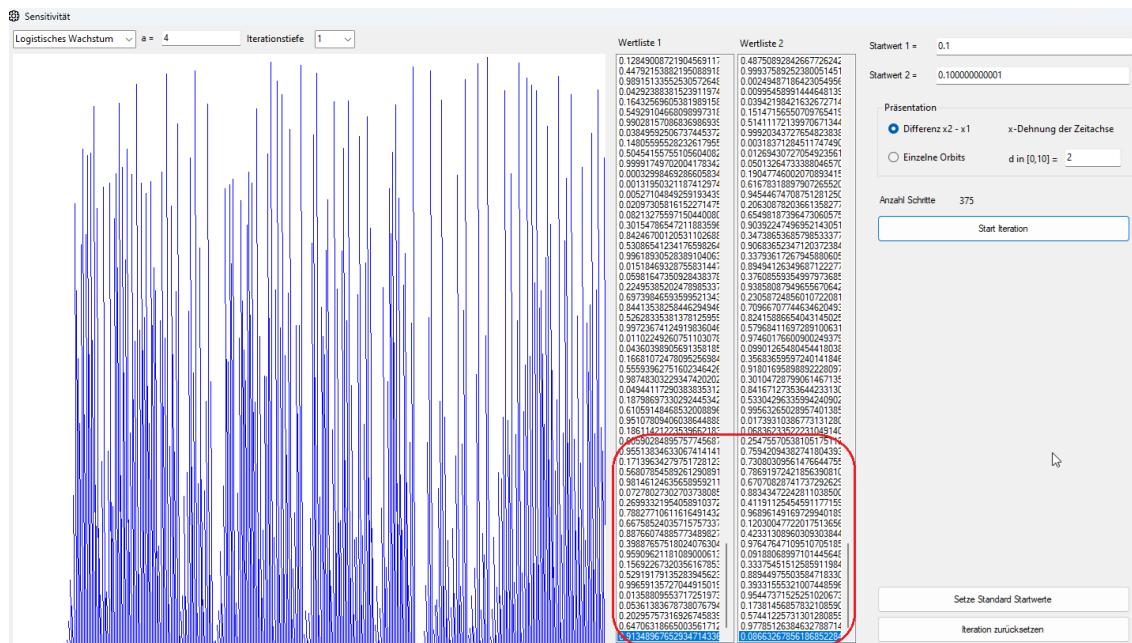
6

Hier werden die Folgenglieder der Iteration ausgehend vom zweiten Startwert aufgelistet.

7, 8

Hier können zwei Startwerte eingegeben werden. Diese können sehr nahe beieinanderliegen. Das Ziel ist dann zu zeigen, dass die erzeugten Folgenglieder bei der Iteration beliebig auseinanderlaufen.

Beispiel



Sensitivität: Zwei leicht verschiedene Startwerte. Angezeigt wird die Differenz der Orbits.

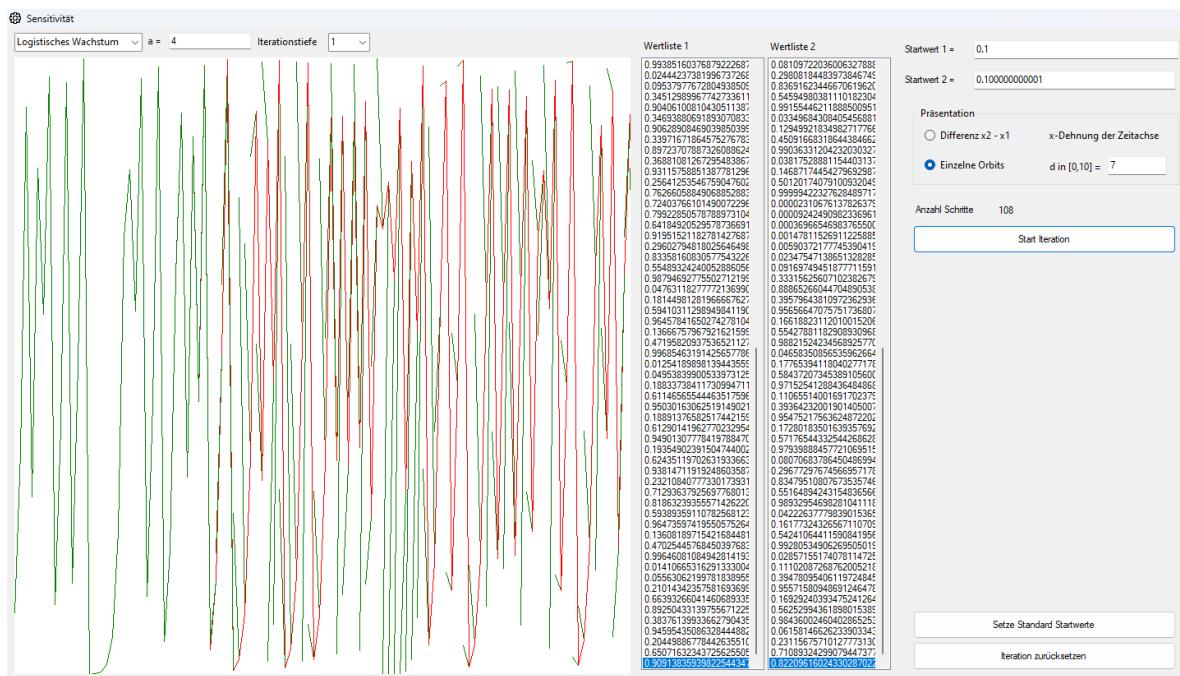
Die beiden Startwerte oben unterscheiden sich um 0.000000000001. Wie man oben sieht, sind die Orbita am Anfang recht ähnlich, beginnen dann aber ab dem rot markierten Bereich allmählich auseinanderzulaufen. In der Grafik wird die Differenz der Orbita dargestellt. Die ist am Anfang fast Null, macht dann aber bald einmal deutliche Sprünge.

10

Für die grafische Anzeige der Iteration stehen zwei Darstellungen zur Verfügung. Entweder wird die Differenz der erzeugten Folgenglieder ausgehend von Startwert eins bzw. zwei angezeigt. Man sieht, dass diese Differenz bereits nach wenigen Schritten innerhalb der Iterationsintervalls sehr gross werden kann.

Alternativ können auch die beiden erzeugten Folgen getrennt angezeigt werden.

In beiden Fällen erfolgt die Darstellung auf der Zeitachse und kann in x-Richtung gedehnt werden. Der Standard ist 2; möglich ist ein Wert zwischen 1 und 10.



Sensitivität: Dieselben Startwerte wie vorhin. Die Orbita werden einzeln angezeigt

In beiden Fällen sieht man, dass die Orbita am Anfang gleichlaufen, aber bald einmal deutlich auseinanderdriften. Zur besseren Sichtbarkeit wurde die Dehnung = 7 gesetzt.

9

Anzeige der Anzahl Iterationsschritte.

11

Beim Drücken der Taste «Start Iteration» wird die Iteration gestartet.

12

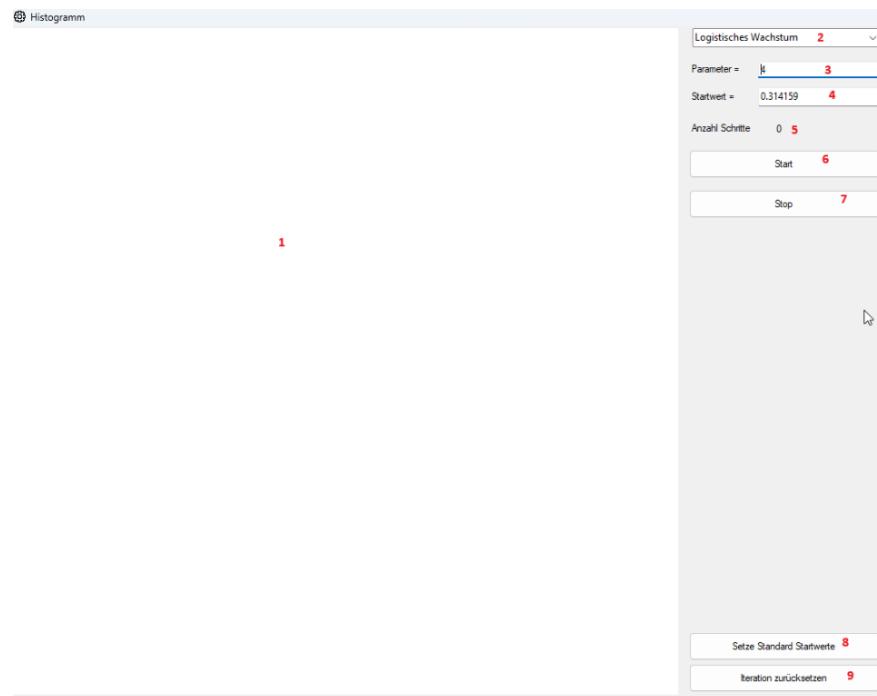
Hier wird die Iteration zurückgesetzt und die Startwerte auf den Standard gesetzt.

13

Hier wird die Iteration zurückgesetzt.

4.3. Histogramme

Im chaotischen Fall kann die Verteilung der Iterationswerte in einem Histogramm untersucht werden (im nicht-chaotischen Fall ist dies nicht interessant). Im Menüpunkt «Wachstumsmodelle - Histogramm» wird folgendes Fenster geöffnet:



Fenster zum Erstellen von Histogrammen

In diesem Fenster wird der Wertebereich in kleine Intervalle aufgeteilt. Die Breite eines solchen Intervalls entspricht etwa der Pixelgrösse. Dann wird gezählt, wie oft ein Intervall im Laufe der Iteration von einem x-Wert «getroffen» wird. In der Mitte oder auch bei der Zeltabbildung ist die Häufigkeitsverteilung dann ziemlich gleichmässig. Beim logistischen Wachstum oder der Parabel sind die Bögen ganz links und ganz rechts erklärbar (siehe mathematische Dokumentation).

1

Hier wird das Histogramm angezeigt.

2

Hier wird die Iterationsfunktion ausgewählt. Zur Verfügung stehen wie immer:

- Zeltabbildung
- Logistisches Wachstum
- Parabel

3

Hier wird der Parameter der Iterationsfunktion eingegeben. Die Untersuchung des Histogramms ist nur im chaotischen Fall sinnvoll. Das ist je nach Iterationsfunktion:

- Zeltabbildung $a = 2$
- Logistisches Wachstum $a = 4$
- Iteration der Parabel $a = 2$

4

Hier kann man einen Startwert für die Iteration eingeben. Wenn der Startwert geändert wird, wird eine bestehende Iteration zurückgesetzt.

5

Anzeige der Schrittzahl während der Iteration.

6

Hier wird die Iteration gestartet

7

Und hier gestoppt.

8

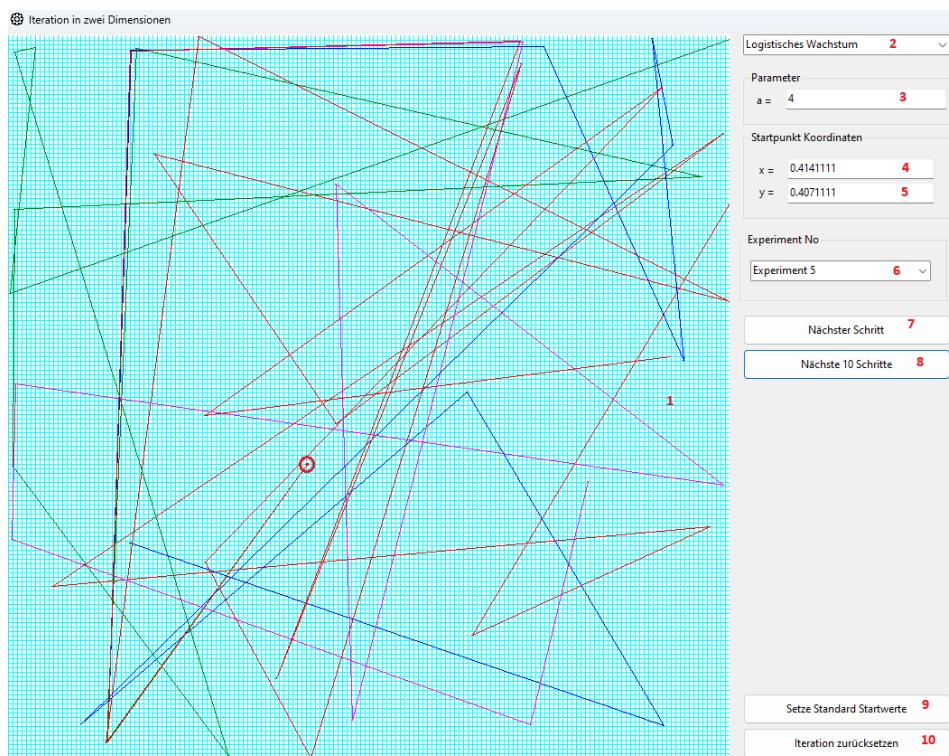
Hier wird die Iteration zurückgesetzt und die Startwerte werden auf den Standard gesetzt.

9

Die Iteration wird zurückgesetzt und die Startwerte bleiben stehen.

4.4. Zweidimensionale Darstellung

Beim Menüpunkt «Wachstumsmodelle - Zweidimensionale Darstellung» wird folgendes Fenster geöffnet:



Fenster für die zweidimensionale Darstellung

Bevor wir die Bedeutung dieses Experimentes erläutern, zuerst die Funktion der einzelnen Felder im obigen Fenster.

1

Der Bereich für die graphische Darstellung

2

Hier wird die Iterationsfunktion ausgewählt. Zur Verfügung stehen wie immer:

- Zeltabbildung
- Logistisches Wachstum
- Parabel

3

Hier wird der Parameter der Iterationsfunktion eingegeben. Die Untersuchung ist nur im chaotischen Fall sinnvoll. Das ist je nach Iterationsfunktion:

- Zeltabbildung $a = 2$
- Logistisches Wachstum $a = 4$
- Iteration der Parabel $a = 2$

4 und 5

Hier werden zwei Startwerte x_1 und y_1 eingetragen. Das liefert einen Startpunkt $P(x_1, y_1)$. Dieser wird links im Diagramm als Punkt eingetragen.

6

Der Experimentator kann nun mit verschiedenen Startpunkten bis zu fünf verschiedene Experimente durchführen. Die entsprechenden Orbits werden mit unterschiedlichen Farben im Diagramm dargestellt. Im Feld 5 kann der Experimentator jedem Experiment eine Nummer zwischen 1 bis 5 zuordnen.

Wir nehmen nun an, dass der Experimentator eine beschränkte Messgenauigkeit hat, welche im Diagramm durch das hellblaue Raster dargestellt wird. Die Rastergrösse beträgt hier 5x5 Pixel, was etwa einem mathematischen Raster von 0.00825×0.00825 Einheiten entspricht.

Der Experimentator hat nun im der oben dargestellten Diagramm 5 Experimente gestartet mit den Startpunkten: $P_1(0.414, 0.407)$, $P_2(0.4141, 0.4071)$, $P_3(0.41411, 0.40711)$,

$P_4(0.414111, 0.407111)$, $P_5(0.4141111, 0.4071111)$

Alle Startpunkte liegen aber im selben Messquadrat des Experimentators. Das heisst, für ihn ist der Startpunkt immer derselbe. Im Diagramm ist dieser durch einen roten Kreis markiert.

Nun lässt er jedes Experiment laufen. Wegen der Sensitivität laufen die Orbits schon nach wenigen Schritten auseinander. Für den Experimentator sieht es also so aus, wie wenn derselbe Startpunkt bei der Iteration zu ganz unterschiedlichen Orbits führt. Das Verhalten des Systems scheint zufällig.

Iteriert wird dabei eine der zur Verfügung stehenden Iterationsfunktionen, und zwar in beiden Komponenten x und y dieselbe.

7

Es wird ein Einzelschritt der Iteration durchgeführt und der neue Punkt im Diagramm mit dem vorhergehenden verbunden.

8

Hier werden die nächsten 10 Schritte der Iteration durchgeführt und im Diagramm nachgetragen.

9

Hier wird die Iteration zurückgesetzt und die Startparameter werden auf den Standard gesetzt.

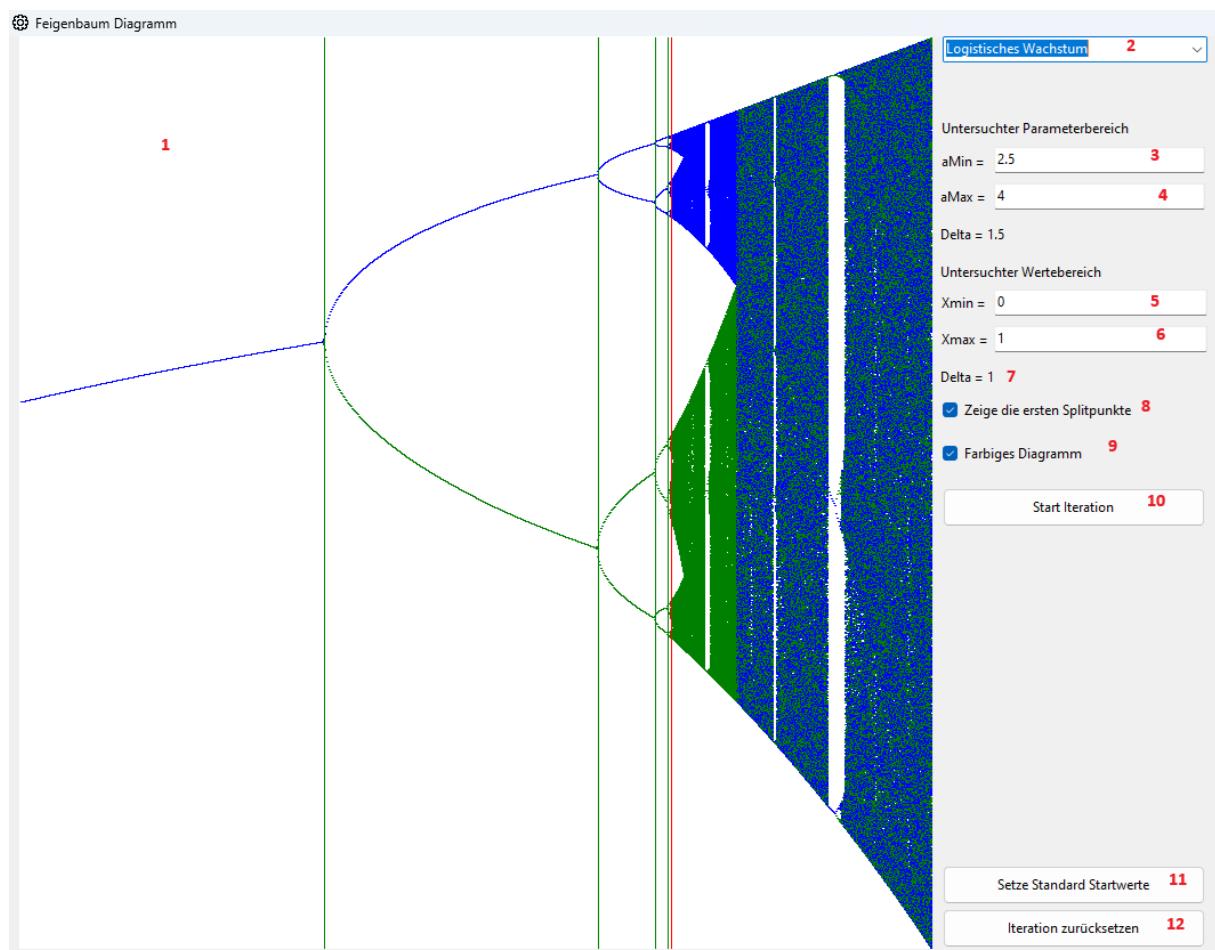
10

Hier wird nur die Iteration zurückgesetzt.

4.5. Feigenbaum Diagramm

Mit dem Simulator kann man das bekannte Feigenbaum-Diagramm und die Periodenverdoppelung untersuchen. Dies wird in der mathematischen Dokumentation näher beschrieben.

Im Menüpunkt «Wachstumsmodelle - Feigenbaum» wird das entsprechende Fenster angezeigt.



Fenster zur Untersuchung des Feigenbaum Diagrammes

Das Diagramm zeigt in horizontaler Richtung Werte des Parameters a , welcher in dieser Richtung linear wächst. Für jeden Parameterwert wird die Iteration zuerst eine Weile lang laufen gelassen in der Hoffnung, dass sie sich bis dann auf einen attraktiven Fixpunkt oder einen attraktiven Zyklus eingependelt hat, wenn es überhaupt einen solchen gibt. Anschliessend wird der Orbit ins Diagramm in vertikaler Richtung eingetragen.

Im obigen Fenster startet die Untersuchung bei $a = 2.5$. Bis zum ersten Splitpunkt (die erste grüne vertikale Linie) hat man einen attraktiven Fixpunkt und die Iteration zeichnet für jedes a in diesem Bereich den Fixpunkt, zu welchem die Iteration konvergiert. Nach dem ersten Splitpunkt hat man einen 2-er Zyklus bis zum zweiten Splitpunkt, dann einen 4-er Zyklus und so weiter, bis das Verhalten chaotisch wird. Dieser Punkt ist mit einer roten vertikalen Linie markiert. Weiter rechts sind mehrheitlich keine Zyklen mehr zu erkennen außer in einzelnen kleinen Fenstern.

Nun zuerst zur Bedeutung der einzelnen Felder.

1

Der Bereich für die graphische Darstellung.

2

Hier kann man die Art der Iteration auswählen. Zur Verfügung stehen:

- Zeltabbildung
- Logistisches Wachstum
- Parabel
- Mandelbrot Iteration für reelle Zahlen

3, 4

Hier wird der zu untersuchende Parameterbereich für den Parameter a festgelegt. In 3 wird das minimale a und in 4 das maximale a festgelegt. Untersucht wird das Parameterintervall $[aMin, aMax]$. Als Delta wird die Breite dieses Bereiches angezeigt.

Diese Werte kann man manuell ändern, sie müssen aber im erlaubten Parameterbereich der Iteration liegen:

- Zeltabbildung: a muss im Bereich $]0,2]$ sein
- Logistisches Wachstum: a muss im Bereich $]0,4]$ sein
- Parabel: a muss im Bereich $]0,2]$ sein
- Mandelbrot: a muss im Bereich $[-2, 0[$ sein

5, 6, 7

Vielleicht möchte man nur einen Ausschnitt des Wertebereichs untersuchen. Dieser Ausschnitt wird hier festgelegt: In 5 den minimalen Wert von x und in 6 den maximalen Wert. Untersucht wird der Wertebereich $[Xmin, Xmax]$. Die Breite des untersuchten Wertebereichs wird in 7 dargestellt.

8

Die vertikalen Geraden im Diagramm zeigen die ersten Splitpunkte der ersten Periodenverdoppelung ab $a \approx 3.5$. Diese Anzeige kann ein- oder ausgeblendet werden.

9

Das Diagramm kann ein- oder zweifarbig dargestellt werden. Diese Option kann hier ausgewählt werden. Wer gerne mit weiteren Farben experimentiert, kann den Programmcode bzw. die Funktion *SetColor* im Code der *FrmFeigenbaum* selbst entsprechend anpassen.

10

Hier wird die Iteration gestartet.

11

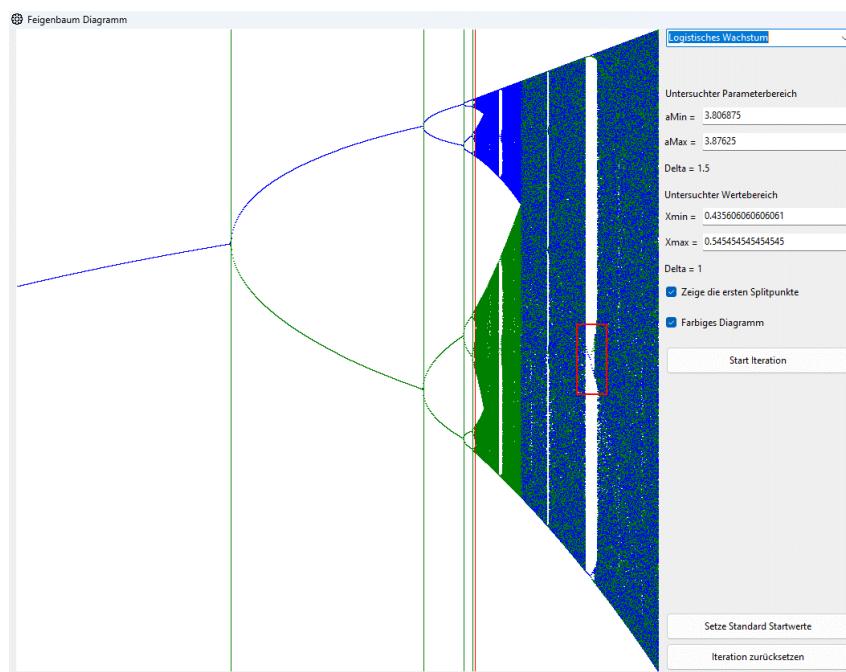
Die Iteration wird zurückgesetzt und die Startparameter auf den Standard gesetzt.

12

Die Iteration wird zurück gesetzt.

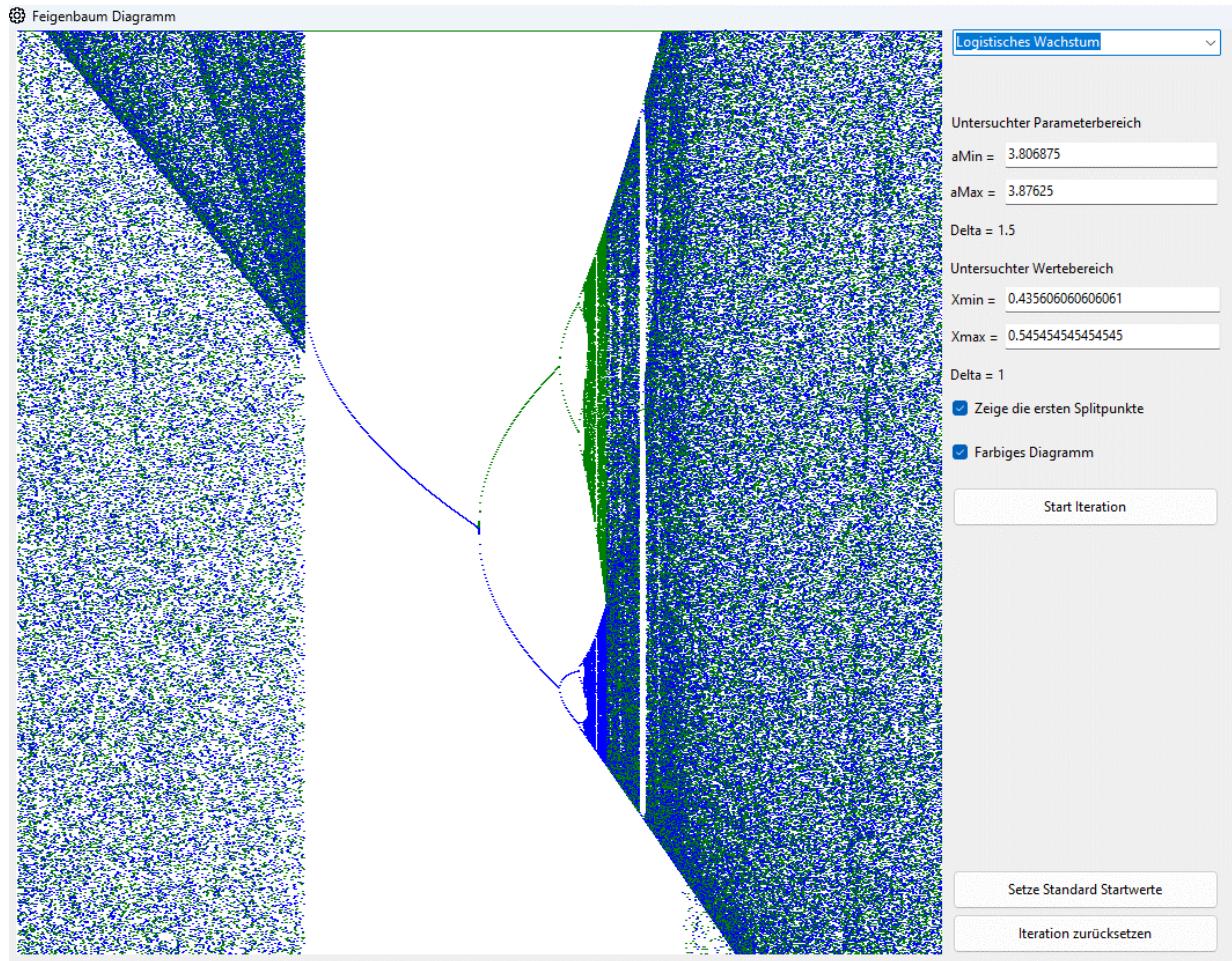
Auswahl von kleinen Ausschnitten im Diagramm

Der Benutzer hat nun die Möglichkeit, einen Ausschnitt, den er genauer untersuchen will, mit Hilfe eines Auswahl-Rechteckes auszuwählen. Das geschieht mit gedrückter linker Maustaste. Dabei wird rechts in den Feldern 2-4 der Parameterbereich und in den Feldern 5-7 der Wertebereich automatisch entsprechend nachgefahren.



Wahl eines kleinen Ausschnittes im Diagramm – siehe rotes Rechteck

Oben möchte der Benutzer die Umgebung eines 3-zyklischen Punktes genauer untersuchen. Dazu hat er mit gedrückter linker Maustaste ein entsprechendes Rechteck gezeichnet. Auf der rechten Seite wurden Parameterbereich und Wertebereich entsprechend angepasst. Wird die Iteration nun gestartet, dann wird dieses Bereich angezeigt.

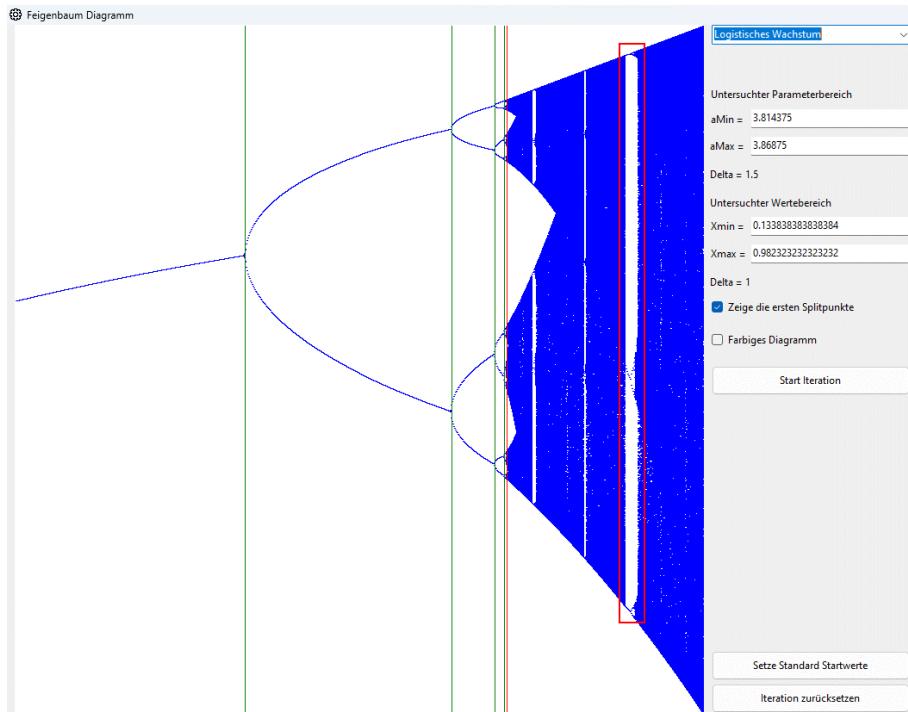


Anzeige der ausgewählten Umgebung des 3-zyklischen Punktes

Man sieht hier, wie aus dem Chaos ein 3-er Zyklus entsteht. Wegen dem reduzierten Wertebereich sieht man hier nur einen Punkt dieses Zyklus. Über eine erneute Periodenverdoppelung geht dann das Diagramm wieder ins Chaos über.

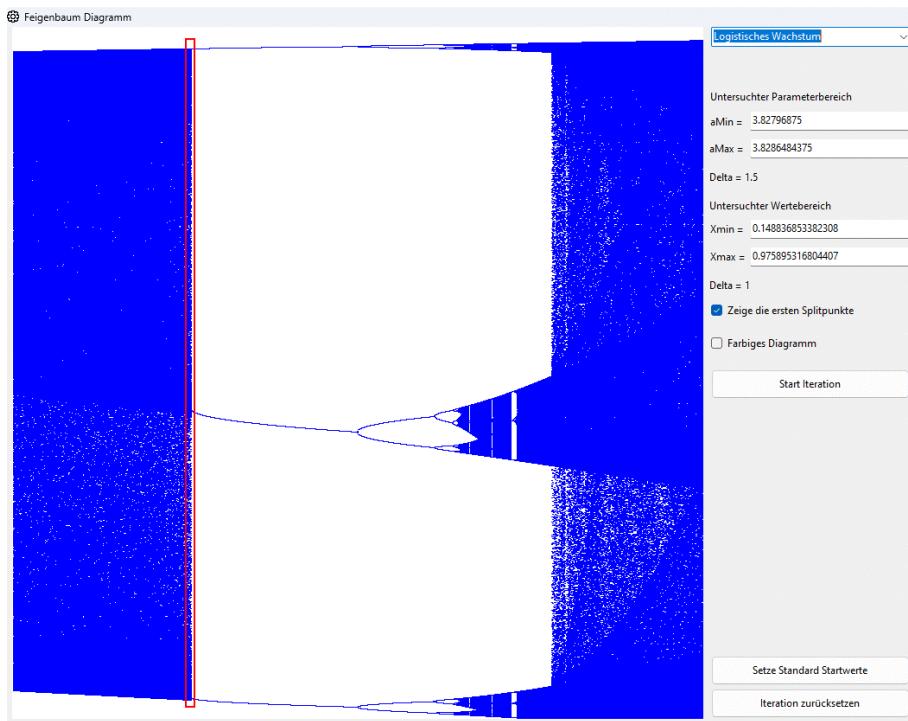
Mit diesem Auswahlverfahren lässt sich auch untersuchen, für welche Parameterwerte von a man bestimmte Zyklen erwarten kann. Durch die Platzierung des Auswahlrechtecks werden nämlich auf der rechten Seite die entsprechenden Intervalle $[a_{\text{Min}}, a_{\text{Max}}]$ angezeigt.

Mit folgendes Auswahldreieck kann man das Auftreten des 3-er Zyklus eingrenzen. Hier kann man ablesen: $a \in [3.806, 3.876]$



Platzierung des Auswahldreiecks zum Bestimmen eines Parameterintervalls

Wenn wir nun die Iteration in diesem Bereich von a wieder starten, können wir der 3-Zyklus weiter eingrenzen:



Der 3-er Zyklus tritt für $a \in [3.8279, 3.8286]$ auf

5. Menü komplexe Iteration

5.1. Newton Iteration

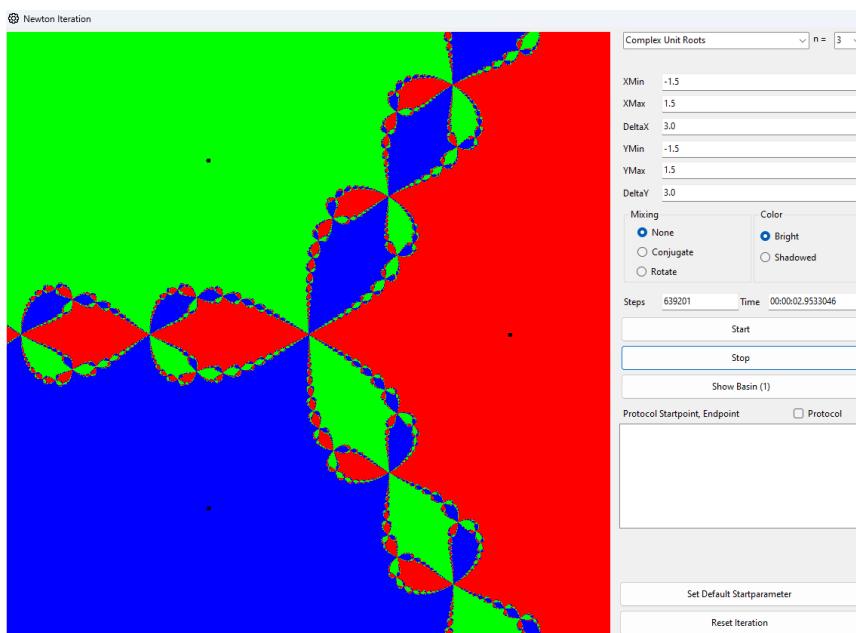
Das Verfahren der Newton Iteration für die Approximation von Nullstellen reeller Funktionen kann in der Mittelschule auf elementarer Basis hergeleitet oder mindestens anschaulich begründet werden. Dieses Verfahren kann auf die komplexe Ebene erweitert werden. Das wird in der mathematischen Dokumentation beschrieben. Dabei geht es weniger darum, effektiv Nullstellen von komplexen Polynomen zu finden, sondern darum, deren «Einzugsbereich» oder «Bassin» zu bestimmen. Damit ist folgendes gemeint: Nehmen wir an, $\zeta \in \mathbb{C}$ sei eine Nullstelle eines komplexen Polynoms $p(z)$. Geben wir dieser Nullstelle die Farbe rot. Wir starten dann die Newton Iteration mit einem Pixelpunkt einer Windowsform bzw. dem dazugehörigen Startpunkt $z_0 \in \mathbb{C}$. Wenn dieser Punkt im Laufe der Iteration gegen die Nullstelle ζ konvergiert, färben wir ihn ebenfalls rot. z_0 liegt dann im Bassin von ζ . Bei mehreren Nullstellen geben wir jeder Nullstelle eine Farbe und färben konvergierenden Startpunkte dann mit der jeweiligen Farbe. Punkte, welche nach einer gewissen Anzahl Schritten kein Konvergenzverhalten zeigen oder gegen ∞ gehen, bleiben schwarz gefärbt.

Ein Punkt gilt als konvergent gegen eine Nullstelle ζ , wenn er dieser genügend nahekommt. Da eine Nullstelle einem superattraktiven Fixpunkt des Newton Verfahrens entspricht, kann sie dann seinem unmittelbaren Bassin nicht mehr «entfliehen».

Bei den Nullstellen des Polynoms $p(z) = z^2 - 1$ streben alle Startpunkte z_0 mit $Re(z_0) < 0$ gegen die Nullstelle -1 und alle Startpunkte mit $Re(z_0) > 0$ gegen die Nullstelle $+1$. Der Rand, welcher die beiden Bassins trennt, ist die y-Achse. Siehe die Herleitung in der mathematischen Dokumentation.

Die Bassins und ihr Rand können bei Polynomen vom Grad > 2 aber sehr komplex aussehen.

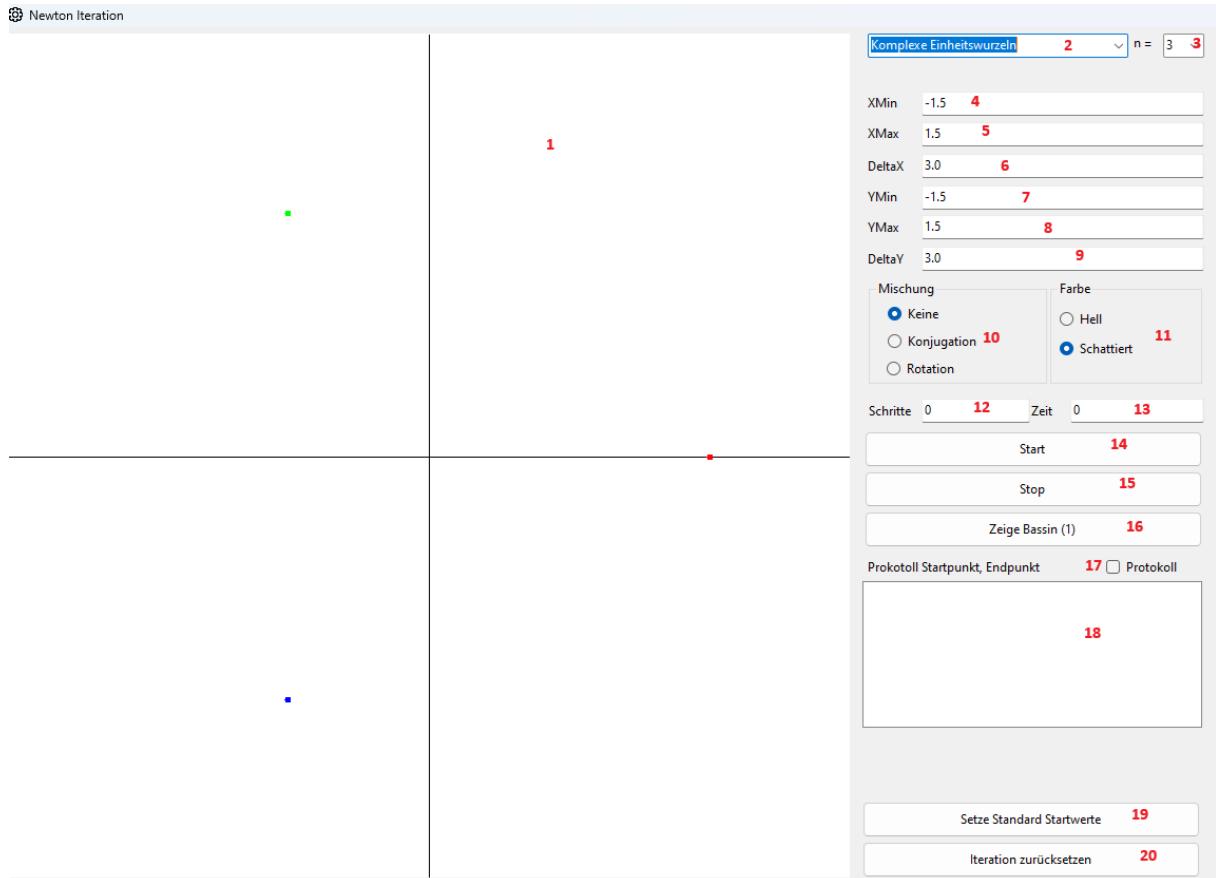
Im Programm können u.a. die komplexen Einheitswurzeln, das sind die Nullstellen des Polynoms $p(z) = z^n - 1, n \in \mathbb{N}$ und ihre Bassins untersucht werden. Im Programm ist n eine natürliche Zahl zwischen 2 und 12. Folgendes Bild zeigt die Bassins der Einheitswurzeln im Falle $n = 3$: $\zeta_1 = 1$ (rot), $\zeta_2 = 0.5 + \frac{\sqrt{3}}{2}i$ (grün), $\zeta_3 = 0.5 - \frac{\sqrt{3}}{2}i$ (blau).



Bassins der Einheitswurzeln vom Grad drei

In der mathematischen Dokumentation werden diese Bassins tiefer diskutiert.

Im Menü «Komplexe Iteration – Newton Iteration» wird folgendes Fenster geöffnet:



1

Zeichenfläche bzw. Ausschnitt der komplexen Ebene.

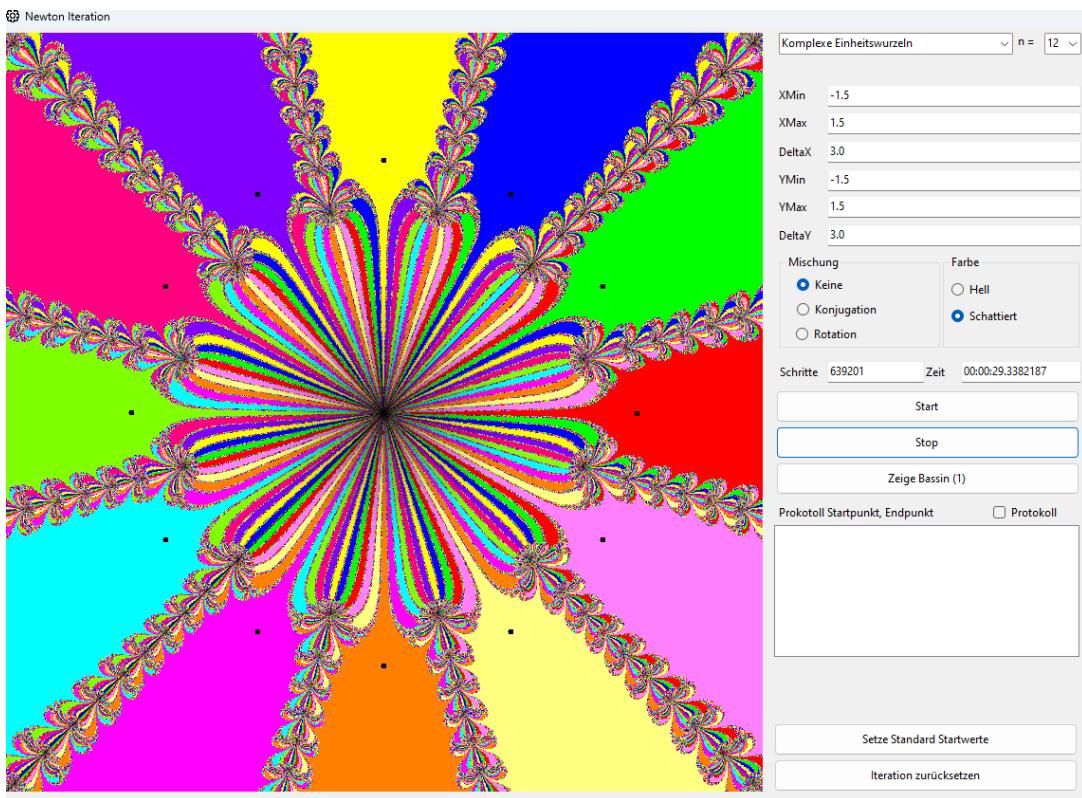
2

Wahl des Polynoms, dessen Nullstellen untersucht werden sollen. Zur Auswahl stehen:

- Komplexe Einheitswurzeln
- Polynom mit drei Nullstellen, nämlich $p(z) = (z - 1)(z + 1)(z - c)$ wobei c durch den Benutzer definiert werden kann
- Komplexe Einheitswurzeln nach der Inversion am Einheitskreis. Damit kann das Verhalten im Unendlichen untersucht werden

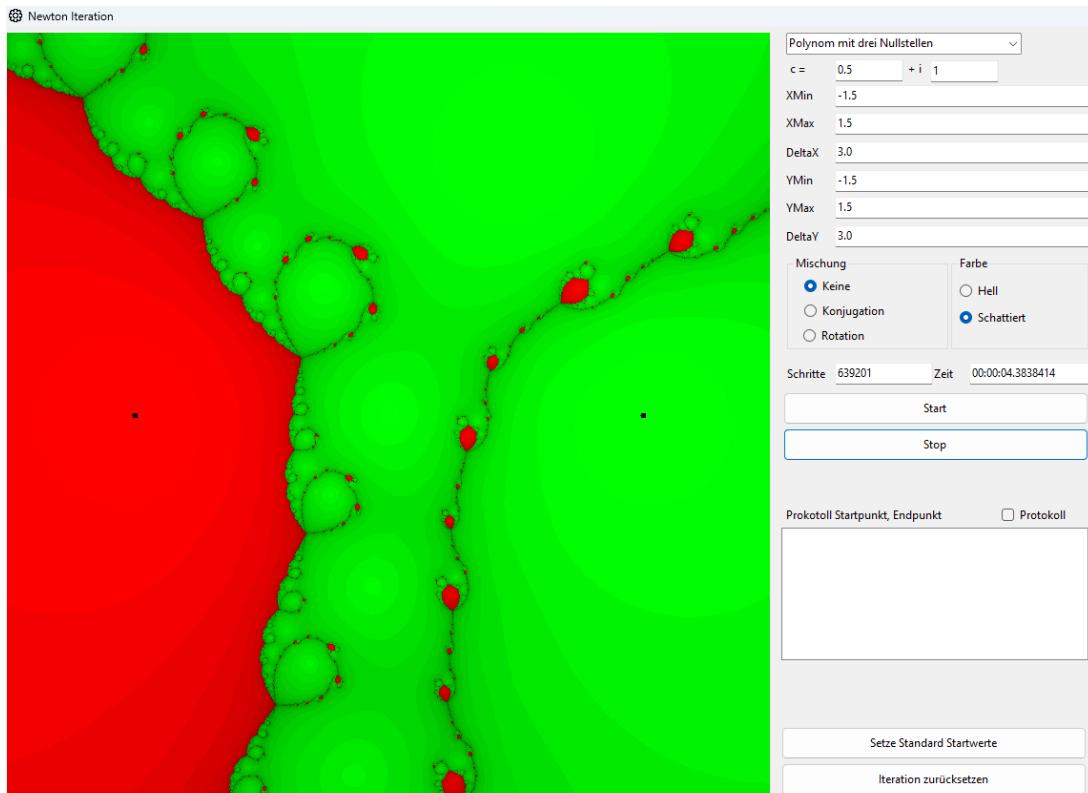
3

Für die komplexen Einheitswurzeln oder deren Invertierte kann hier der Grad festgelegt werden. Möglich sind Werte von n zwischen 3 bis 12.



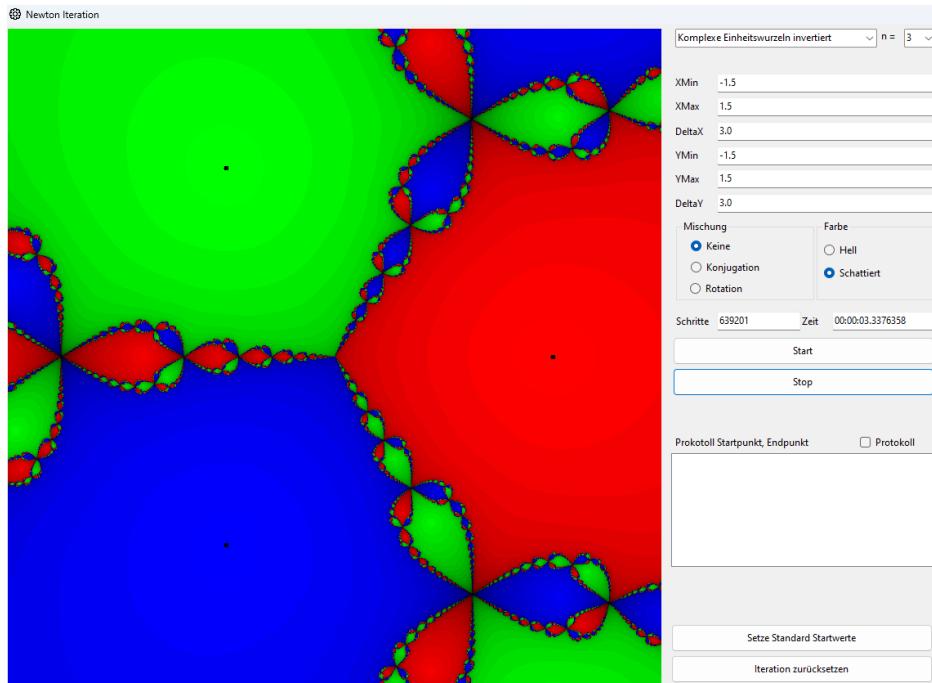
Bassins der zwölften Einheitswurzeln in schattierter Darstellung

Wenn die Option «Polynom mit drei Nullstellen» gewählt wird, erscheint unterhalb der Auswahlbox ein Feld, in welchem c eingegeben werden kann:



Polynom vom Grad drei mit den Nullstellen $\pm 1, 0.5 + i$

Bei den invertierten Einheitswurzeln kann untersucht werden, wie das Newton Verfahren im unendlichen wirkt. Für Details siehe mathematische Dokumentation.

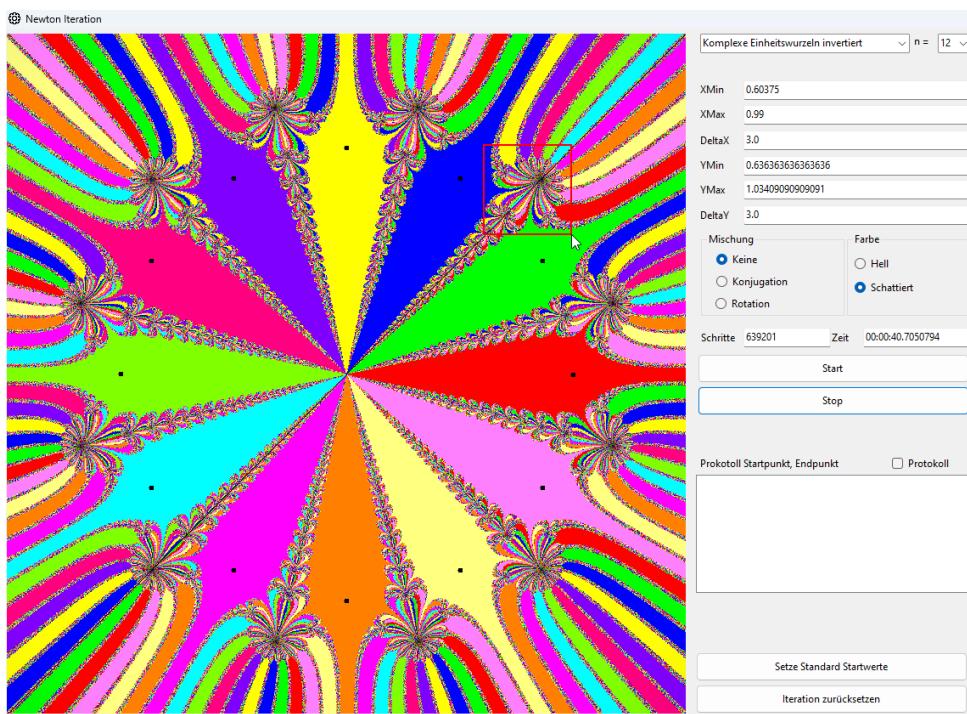


Invertierte Einheitswurzeln vom Grad drei

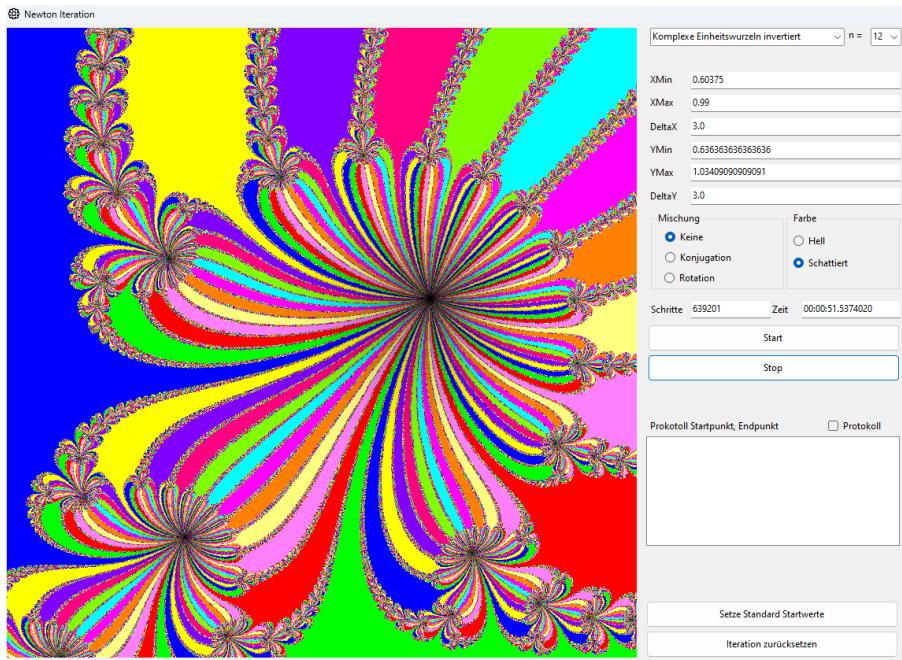
4, 5, 6, 7, 8, 9

Falls ein Ausschnitt der komplexen Ebene mit der Maus ausgewählt wird (durch gedrückte linke Maustaste) werden hier die Koordinaten des Ausschnittes angezeigt inklusive deren Breite und Höhe.

Der Benutzer kann hier auch einen Ausschnitt manuell eingeben.



Invertierte Einheitswurzeln vom Grad 12 mit einem gewählten Ausschnitt



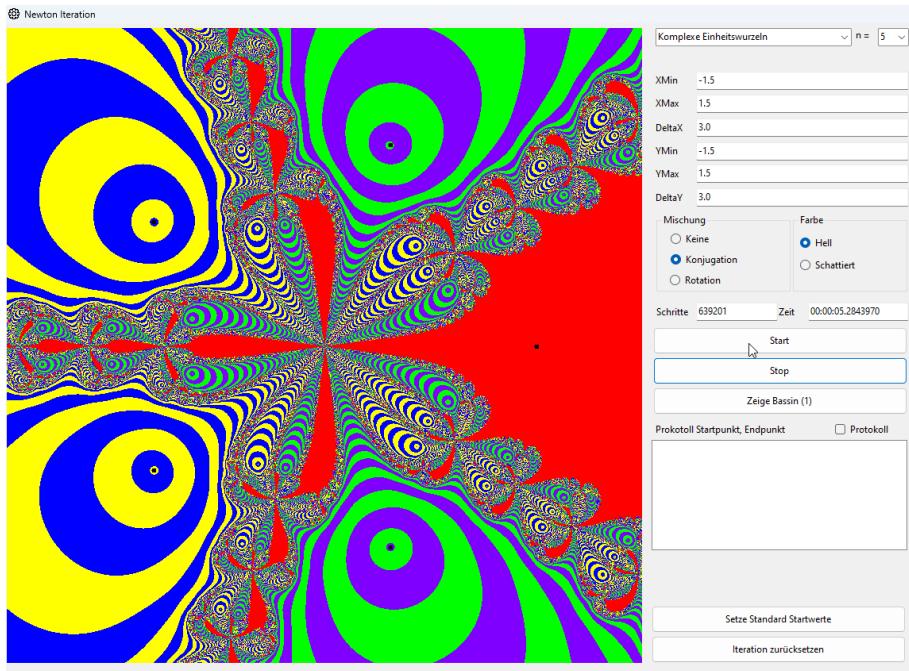
Iteration im gewählten Ausschnitt, der dadurch vergrößert dargestellt wird

Da das Bild fraktale Eigenschaften hat, kann der Zoom innerhalb der Rechengenauigkeit unendlich fortgesetzt werden und es erscheinen immer ähnliche Bilder

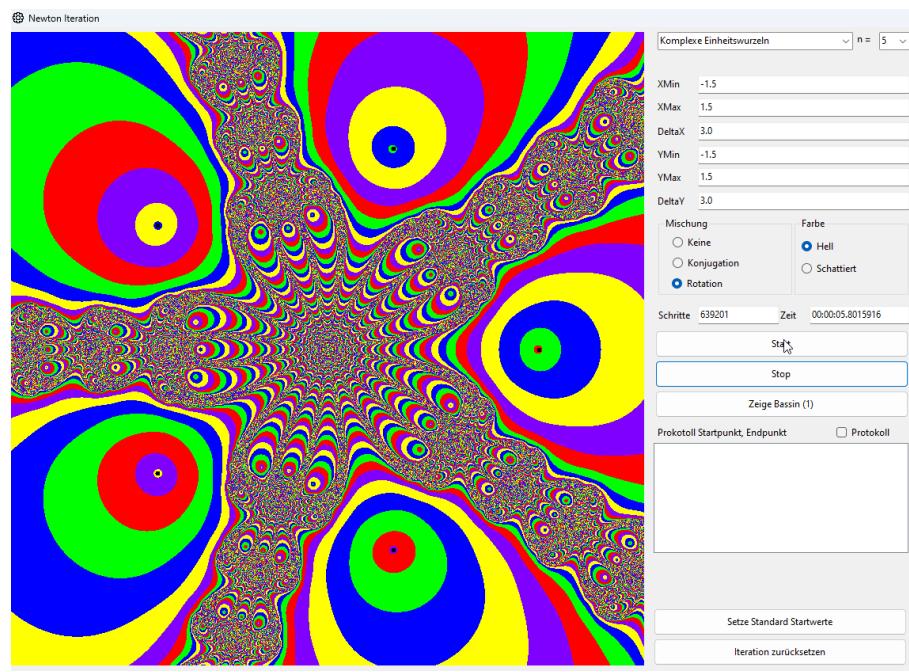
10

Hier geht es eher um die Erzeugung interessanter Bilder. Dabei werden die Bassins «vermischt». Zur Auswahl stehen:

- Keine Mischung: Man sieht die originalen und echten Bassins.
- Konjugation: Hier wird die Iterierte z_n bei jedem Iterationsschritt konjugiert. Das führt dazu, dass Bassins, welche symmetrisch zur x-Achse liegen, bei der Iteration abwechselungsweise «besucht» werden der Startpunkt dann die Farbe des Bassins erhält, bei dem er sich beim Ende der Iteration gerade befindet. Damit wird verdeutlicht, ob ein Punkt nach einer geraden oder ungeraden Anzahl von Schritten bei «seiner» Nullstelle landet.
- Rotation: Dieselbe Idee, aber der Iterationspunkt wird bei jedem Iterationsschritt um $2\pi/n$ gedreht.



Darstellung der fünften Einheitswurzeln inklusive Konjugation



Rotierte Darstellung der Einheitswurzeln vom Grad fünf

Wenn man zum Beispiel vom roten Zentrum der Einheitswurzel $\zeta = 1$ im Uhrzeigersinn durch die Bassins geht, sieht man, dass die Farbe rot bei jeder weiteren Drehung eine Stufe weiter vom Zentrum wegrückt, bis man nach fünf Schritten wieder im Bassin von $\zeta = 1$ landet.

11

Damit die «Schnelligkeit» der Konvergenz besser sichtbar ist, kann hier die Farbe des Bassins noch unterteilt werden. Zur Verfügung stehen:

- Hell: Die Bassinfarbe ist einheitlich hell. Dies ist der Standard, wenn gleichzeitig die Option «Rotation» oder «Konjugation» gewählt wird.

- Schattiert: Punkte, welche langsamer konvergieren, erhalten zwar die Farbe des Bassins, zu welchem sie gehören, aber etwas dunkler als Punkte, welche schnell konvergieren. Dies ist der Standard, wenn die Option «Keine Mischung» gewählt wird.

12, 13

Da die Erzeugung der Bilder Performance kritisch ist, werden hier die Anzahl der Iterationsschritte und die abgelaufene Zeit für die Iteration angezeigt.

14

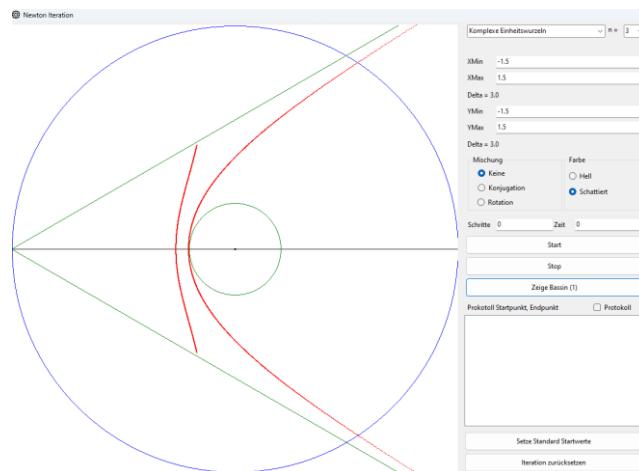
Hier wird die Iteration gestartet oder fortgesetzt.

15

Die Iteration wird gestoppt.

16

Für die Entscheidung, wann ein Iterationspunkt im unmittelbaren Bassin einer Einheitswurzel liegt, muss dieses Bassin bestimmt werden. Diese Taste skizziert das Bassin von 1. Siehe mathematische Dokumentation.



Das unmittelbare Bassin von 1

17

Durch Aktivierung der Checkbox kann ein Protokoll ausgegeben werden. Implementiert ist die Anzeige des jeweiligen Startpunktes sowie ausgehend von ihm der letzte iterierte Punkt, bevor er als konvergent gilt. Allerdings wird durch die Ausgabe eines Protokolls die Iteration wesentlich verlangsamt.

18

Anzeige des Protokolls.

19

Hier wird die Iteration zurückgesetzt und die Auswahlfenster werden auf Standard gesetzt.

20

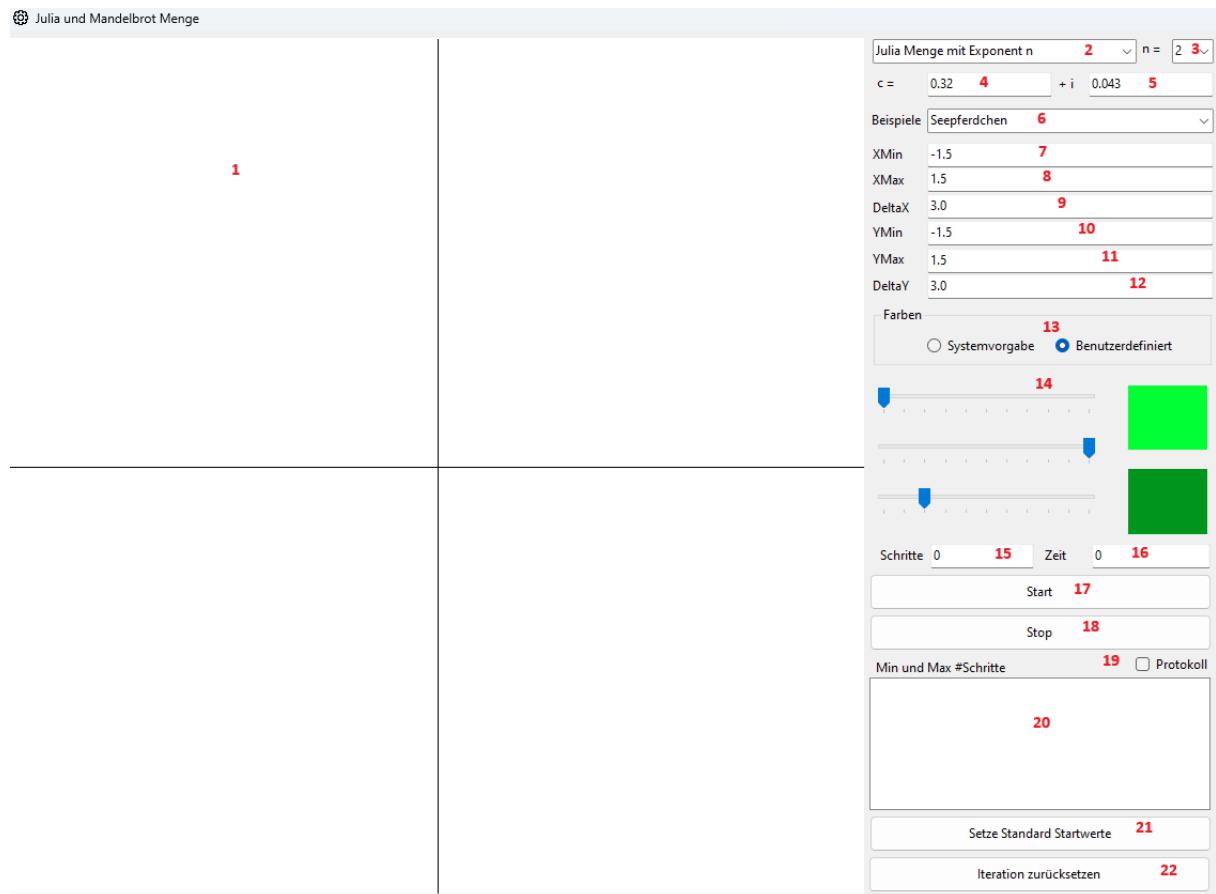
Hier wird nur die Iteration zurückgesetzt.

5.2. Julia Menge und Mandelbrot Menge

Bei der Iteration der Funktion $f(z) = z^2$ streben alle Startpunkte z_0 mit $|z_0| < 1$ gegen 0. Alle Startpunkte mit $|z_0| > 1$ streben gegen unendlich. Die Startpunkte auf dem Einheitskreis bleiben auf dem Einheitskreis. Aber das Verhalten der Iteration auf dem Einheitskreis ist chaotisch.

Nun «baut» man eine kleine Störung ein und ersetzt die Iteration durch $f(z) = z^2 + c, c \in \mathbb{C}$. Dabei kann $c = a + ib; a, b \in \mathbb{R}$ durch den Benutzer eingegeben werden. Die Nullstellen von f sind dann die Wurzeln von $-c$. Es gibt nun Startpunkte z_0 , welche gegen diese Nullstellen konvergieren, also bei denen die erzeugte Punktfolge im endlichen bleibt und andere Startpunkte, welche gegen unendlich streben. Die Trennlinie zwischen den entsprechenden Bassins, welche vorerst ein Kreis war, wird dann aber immer mehr verformt, bis sie sogar zu Staub zerfällt. Diese Trennline, welche gleichzeitig der Rand der Bassins ist, ist eine Julia Menge (siehe mathematische Dokumentation). Der «Simulator» ermöglicht nun die Erzeugung solcher Julia Mengen.

Im Menü «Komplexe Iteration – Julia Menge» kann folgendes Fenster geöffnet werden:



Fenster zur Erzeugung der Julia und Mandelbrot Menge

1

Zeichenfläche und Ausschnitt aus der komplexen Ebene

2

Wahl zwischen Julia- und Mandelbrot Menge mit Exponent n (Iteration von $f(z) = z^n + c$). Bei der Julianamenge untersucht man für jeden Startpunkt $z_0 \in \mathbb{C}$, ob er bei der Iteration gegen ∞ geht. Wenn nicht, wird der Startpunkt schwarz gefärbt. Wenn ja, erhält er eine umso hellere Farbe je mehr

Iterationsschritte es braucht, bevor feststeht, dass er gegen ∞ strebt. Kriterien dazu: Siehe mathematische Dokumentation.

3

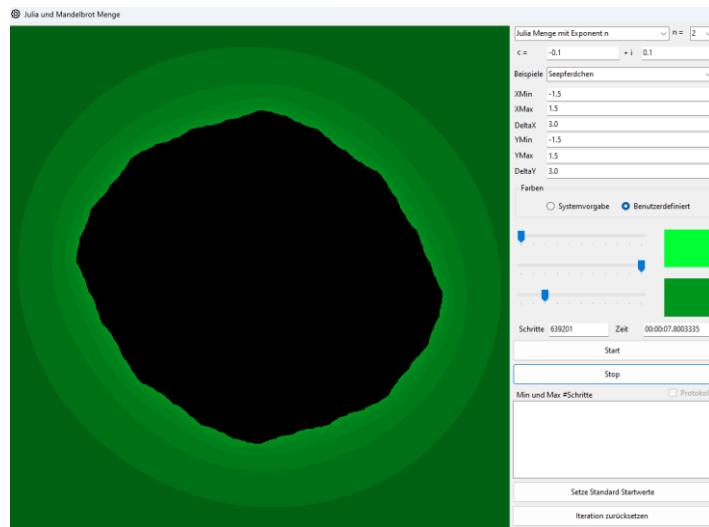
Wahl des Exponenten $n \in \mathbb{N}$.

4, 5

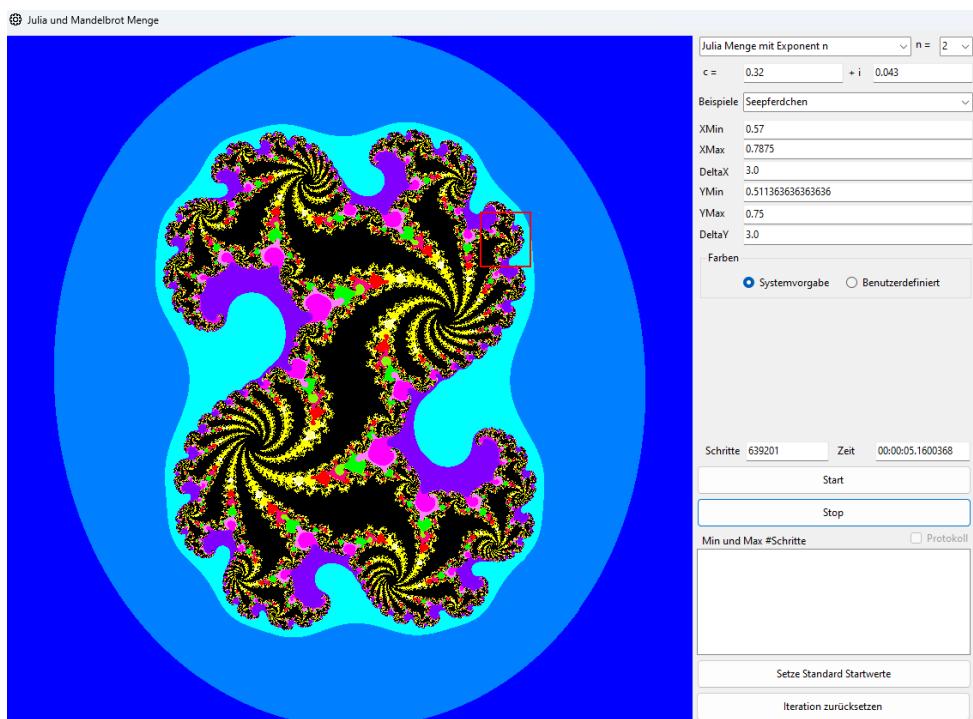
Eingabe der komplexen Konstanten $c = a + ib$.

6

Hier sind einige Beispiele vordefiniert. Bei der Wahl wird die entsprechende Konstante c gesetzt.

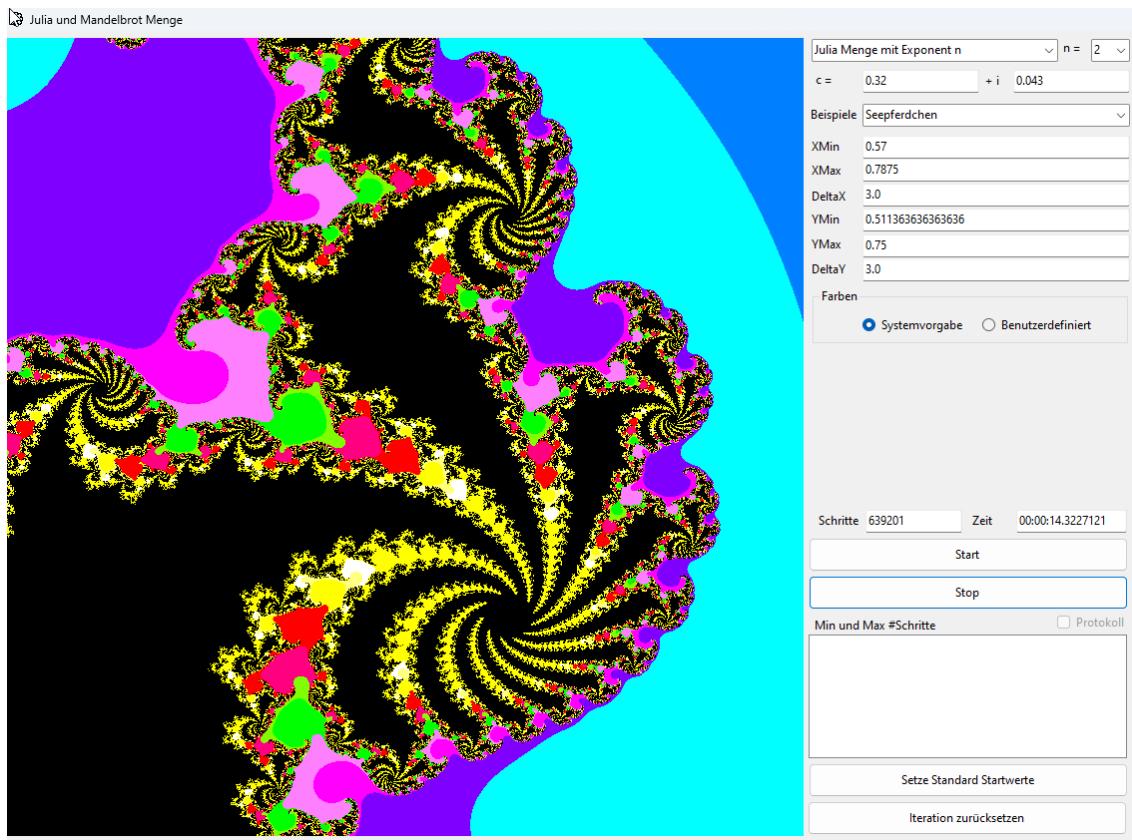


Julia Menge bei einer kleinen «Störung» $c = -0.1 + 0.1i$. Gegenüber $c = 0$ ist der «Einheitskreis» etwas verschrumpelt. Die Farbe ist benutzerdefiniert.

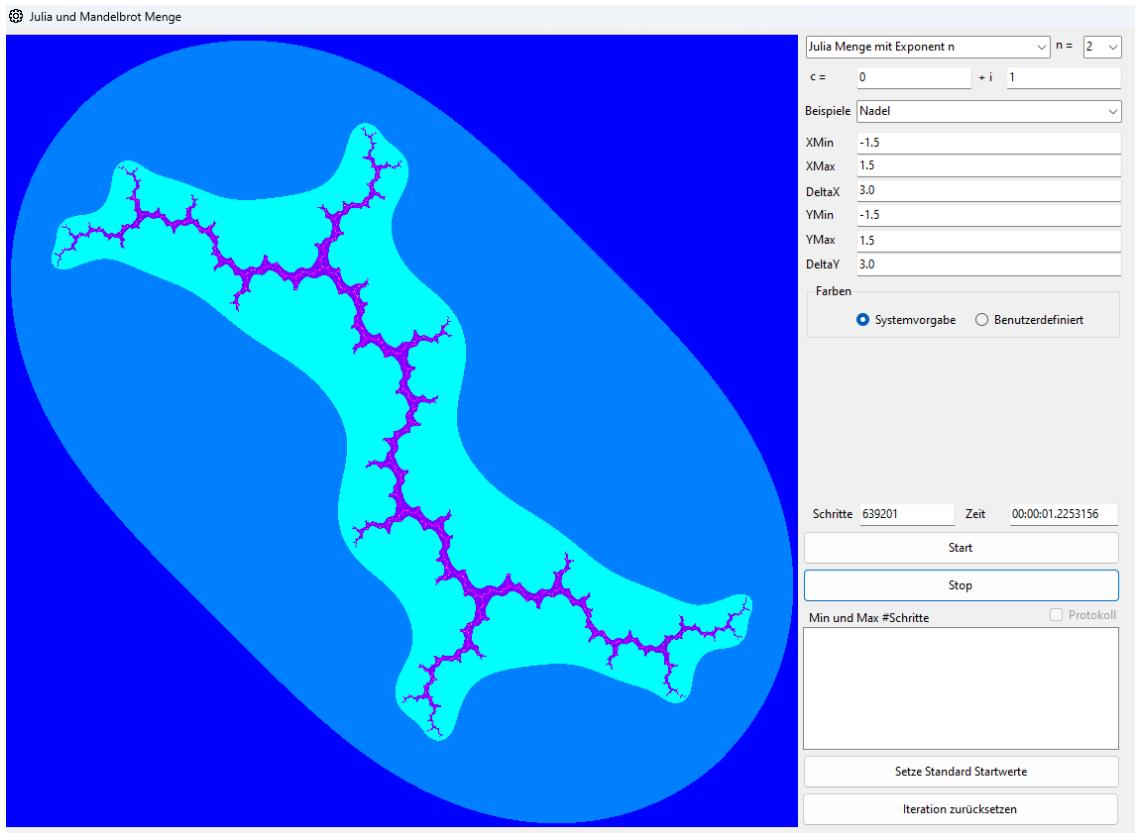


Das «Seepferdchen» mit $c = 0.32 + 0.043i$

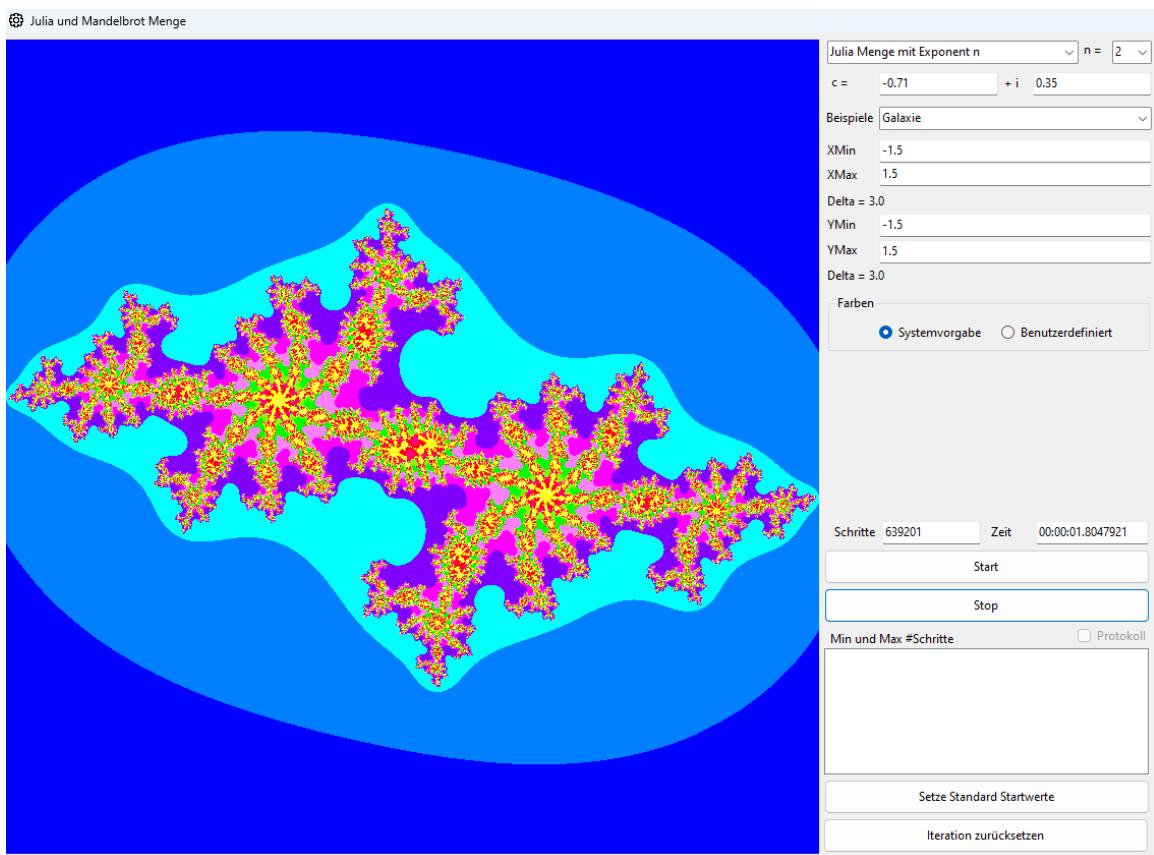
Hier ist die «Störung» grösser. Der Einheitskreis hat sich in ein Fraktal verwandelt und ist nicht wiederzuerkennen. Gewählt wurden die Standardfarben des Systems. Ferner wurde mit gedrückter Maustaste eine Auswahl getroffen, welche nachfolgend untersucht wird.



Das Seepferdchen im gewählten Ausschnitt.



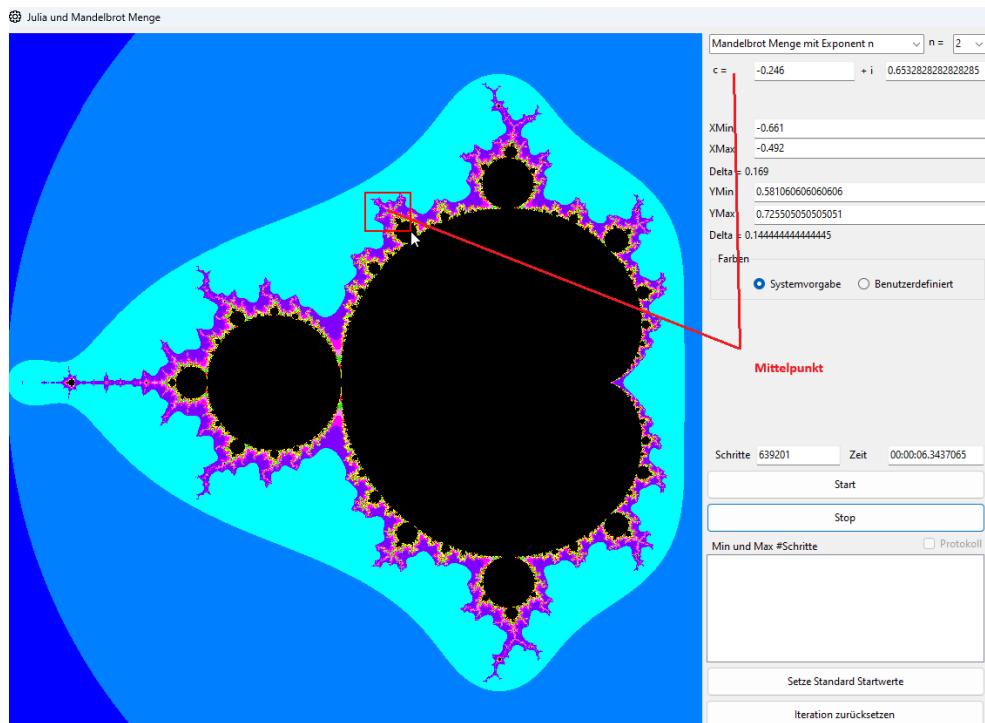
Julia Menge für die «Nadel» mit $c = i$. Sie ist nur noch ein dünnes verästeltes Gebilde.



Die Julia Menge für die «Galaxie» mit $c = -0.71 + 0.35i$

Wie könnte man sich ein Bild darüber machen, für welche c die Julia Menge noch innere Punkte enthält? Darüber gibt die Mandelbrot Menge Auskunft. Es ist nämlich so, dass jedes Bassin eines attraktiven Fixpunktes oder attraktiven Zyklus der Julia Menge mindestens einen kritischen Punkt enthält, also einen Punkt mit $f'(z) = 0$. Im Falle von $f(z) = z^n + c$ ist das der Nullpunkt. Wenn also der Nullpunkt bei der Iteration im Endlichen bleibt, dann existiert mindestens ein Bassin eines attraktiven Zyklus und damit eine Julia Menge mit inneren Punkten. Wenn der Nullpunkt gegen Unendlich strebt, dann existiert kein Bassin und kein attraktiver Fixpunkt. Dann zerfällt die Julia Menge zu «Staub».

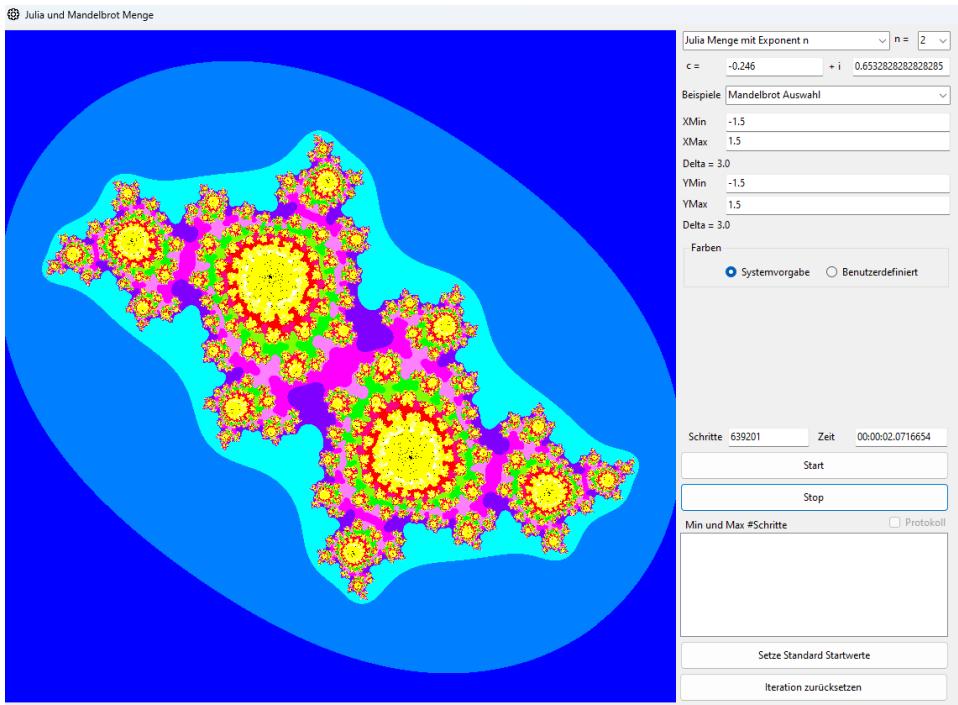
Die Mandelbrotmenge ist nun die «Landkarte» der $c \in \mathbb{C}$ und zeigt das Verhalten des Nullpunktes bei der Iteration. Man wählt also ein c als Parameter der Iteration, wählt dann den Startpunkt $z_0 = 0$ und schaut, ob dieser bei der Iteration gegen ∞ strebt. Wenn nicht, färbt man den Punkt schwarz. Wenn ja, färbt man ihn in einer helleren Farbe, wenn die Divergenz langsamer verläuft. Für weitere Details und die Kriterien dazu: siehe mathematische Dokumentation.



Die Mandelbrot Menge

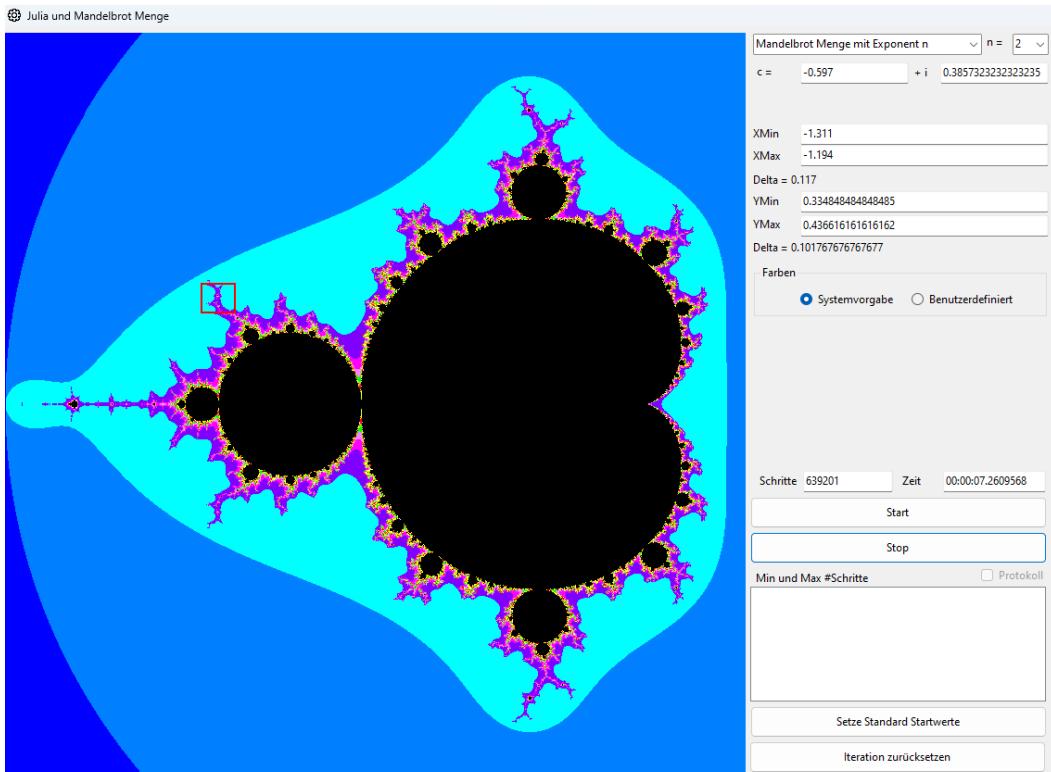
Wenn man mit gedrückter Maustaste einen Teilbereich wählt, werden die entsprechenden Koordinaten in den Feldern XMin, XMax, YMin, YMax angezeigt. Der Wert von c wird dann entsprechend dem Mittelpunkt der Auswahl gesetzt.

Wechselt man jetzt via Combobox rechts oben wieder zur Julia Menge, wird der Wert von c als «Mandelbrot Auswahl» in die Liste der Beispiele aufgenommen. Wenn man diese Option wählt, wird die zu diesem c gehörende Julia Menge erzeugt:

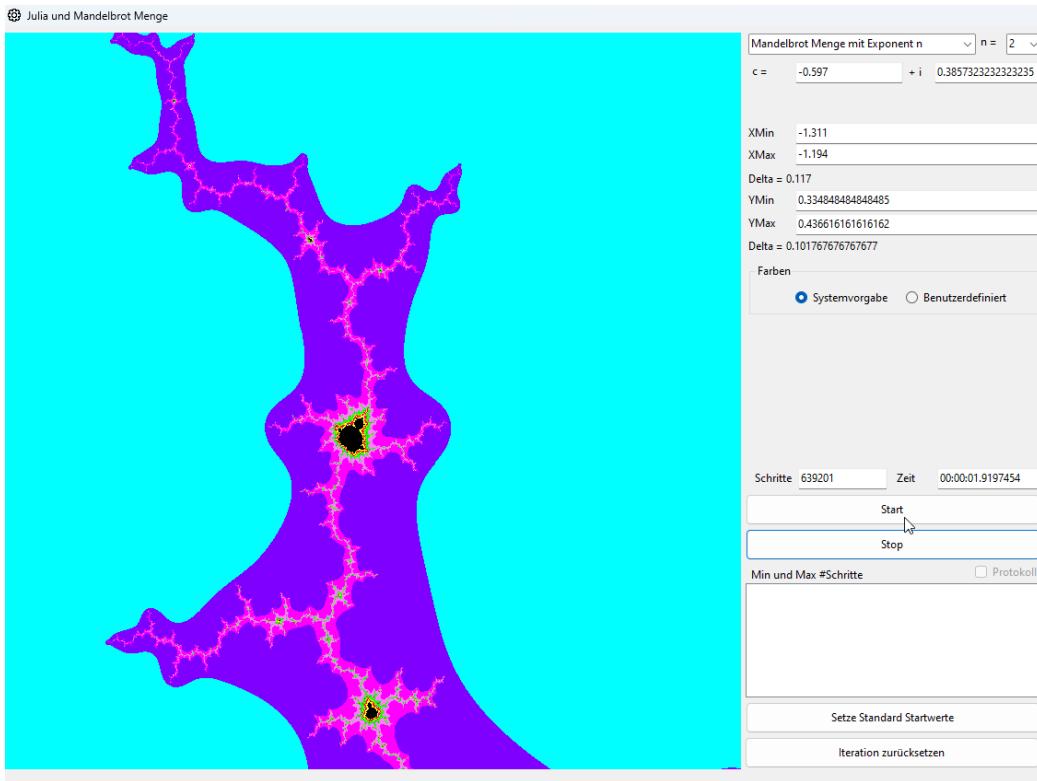


Zum vorherigen c-Wert gehörende Julia Menge

Wie die Julia Menge hat auch die Mandelbrot Menge fraktale Eigenschaften. Das heisst, wenn man einen Ausschnitt des Bildes vergrössert, erscheinen immer wieder ähnliche Bilder.

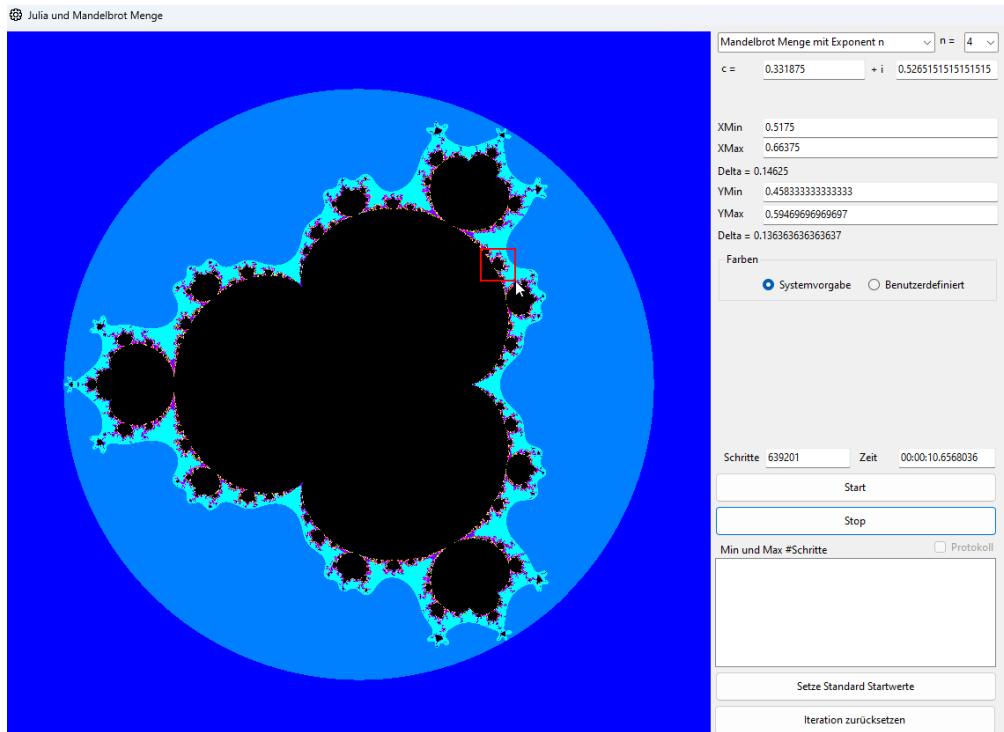


Noch einmal die Mandelbrot Menge mit einem gewählten kleinen Ausschnitt.

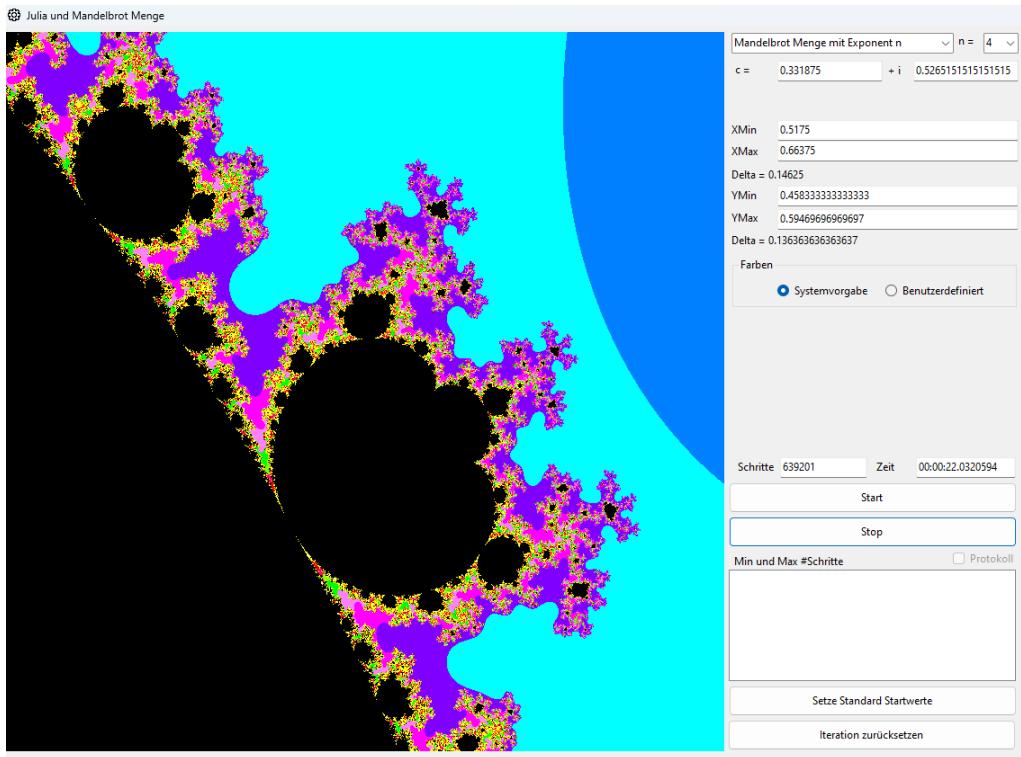


In diesem Ausschnitt erscheinen wieder Exemplare der Mandelbrot Menge

Betreffend der Wahl des Exponenten: n ist eine natürliche Zahl zwischen 2 und 12. Die Fälle $n > 2$ bringen keine grundsätzlich weiteren Erkenntnisse. Sie können aber zur Erzeugung interessanter Bilder verwendet werden. Die Rotationssymmetrie erhöht sich dann entsprechend dem Wert von n . Bei der Julia Menge ist diese n -fach, bei der Mandelbrotmenge $(n-1)$ -fach.



Mandelbrot Menge für $n = 4$ und eine getroffene Auswahl



Das Bild der getroffenen Auswahl

7, 8, 9, 10, 11, 12

Falls im erzeugten Bild ein Ausschnitt mit gedrückter Maustaste gewählt wird, werden hier dessen Koordinaten angezeigt. Da die Mandelbrotmenge eine «Landkarte» für die jeweiligen Julia Mengen ist, wird bei einem Ausschnitt aus der Mandelbrotmenge auch die Zahl c als Zentrum des Ausschnittes gesetzt. Wechselt man dann zur Option «Julia Menge», ist dieses c in der Beispielliste unter «Mandelbrot Auswahl» aufgeführt und man kann die entsprechende Julianenge erzeugen.

13

Als Systemvorgabe wurden geeignete Farbstufen programmiert. Je länger der Startpunkt braucht, um gegen Unendlich zu konvergieren, desto heller die Farbe. Startpunkte, welche bei der Iteration im Endlichen bleiben, sind schwarz.

Der Benutzer kann auch selbst eine Farbe definieren. Anstelle der Farbstufen in Abhängigkeit von der Konvergenzgeschwindigkeit werden dann Schattierungen der entsprechenden Farbstufe angezeigt.

14

Regler für die Zusammensetzung der Benutzerfarbe aus den Grundfarben rot, grün und blau.

15, 16

Anzeige der Anzahl Iterationsschritte und der abgelaufenen Zeit.

17

Start oder Fortsetzung der Iteration.

18

Stopp der Iteration.

19

Durch Aktivierung der Checkbox kann die Ausgabe eines Protokolls verlangt werden. Implementiert sind die jeweiligen minimalen und maximalen Anzahl Iterationsschritte. Durch die Erzeugung eines Protokolls kann die Iterationsgeschwindigkeit sinken.

20

Darstellung des Protokolls.

21

Die Iteration wird zurückgesetzt und die Benutzerparameter werden auf den Standard gesetzt.

22

Die Iteration wird zurückgesetzt.

6. Menü Mechanik

6.1. Numerische Methoden

Grundlage für diese Experimente ist das klassische Federpendel, welches hier in einer Cosinus Schwingung schwingt. Seine Schwingung kann nun verglichen werden mit Federpendeln, welche nicht gemäss Cosinus schwingen, sondern bei denen deren Differentialgleichung durch numerische Methoden angenähert sind. Es handelt sich dabei um einfache Methoden, welche elementar zugänglich sind (ausser vielleicht das Runge Kutta Verfahren). Zur Verfügung stehen:

- Euler Implizit
- Euler Explizit
- Implizite Mittelpunktsregel
- Runge Kutta Verfahren Stufe 4

Diese Methoden werden in der mathematischen Dokumentation im Detail beschrieben.

Zur Untersuchung öffnet man durch das Menü folgende Form:



Fenster zur Untersuchung der numerischen Verfahren

Im linken Bereich werden die Schwingungen der Pendel angezeigt. Grün: Das «echte» Federpendel mit der geschlossenen Lösung seiner Differentialgleichung, welches hier in einer Cosinus-Schwingung schwingt. Rot: Das Federpendel mit einer numerischen Lösung der entsprechenden Differentialgleichung.

Die Pendel sind in der Null-Position. Wenn man die Simulation hier startet, schwingen die Pendel nicht. Sie müssen mit gedrückter linker Maustaste in Ausgangsposition gebracht werden.

Waagrecht: Man sieht die Zeitachse und die Null-Position der Pendel. In y-Richtung ist die Pendelposition im Intervall [-1, 1].

1

Bereich für die graphische Darstellung

2

Hier wird das numerische Verfahren für das rote Pendel ausgewählt. Als Kontrolle kann auch das rote Pendel als «reales Federpendel» gewählt werden.

3

Das reale, grüne Pendel inkrementiert die Zeitvariable in jedem Schritt um 0.1. Das rote Pendel kann dieses Intervall weiter unterteilen entsprechend der in **2** gewählten Schrittweite. Wenn diese z.B. 0.02 beträgt wie oben im Bild, dann führt das rote Pendel 5 Iterationsschritte mit der Schrittweite 0.02 durch, bevor seine Position im Bild angezeigt wird. Das grüne Pendel wird dann gleichzeitig mit der Schrittweite 0.1 angezeigt. Das führt dazu, dass die Pendel synchron sind.

4

Wenn man die Option «gestreckte Darstellung» aktiviert, dann schwingen beide Pendel mit derselben Schrittweite für den Zeitparameter, also z.B. beide Pendel mit der Schrittweite 0.02. Das führt zu einer gestreckten Darstellung für beide Pendel und man sieht die Abweichungen zwischen grünem und rotem Pendel besser.

5

Hier wird die Anzahl der Schritte des grünen Pendels auf der Zeitachse angezeigt.

6

Hier werden die Pendel gestartet. Wird die Schwingung unterbrochen, kann sie mit dieser Taste fortgesetzt werden.

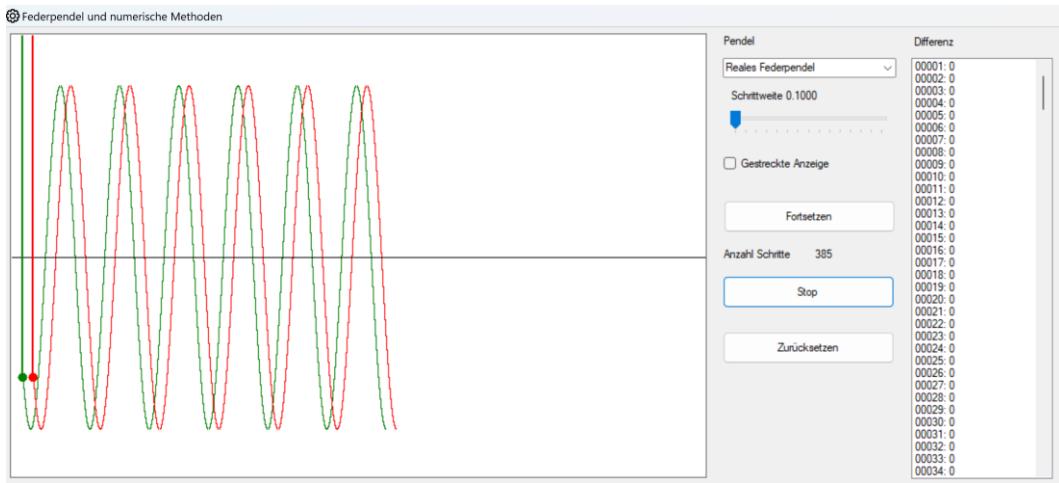
7

Hier wird die Schwingung der Pendel unterbrochen.

8

Hier wird die Differenz der y-Positionen des grünen und roten Pendels protokolliert.

Nachfolgend einige Beispiele.



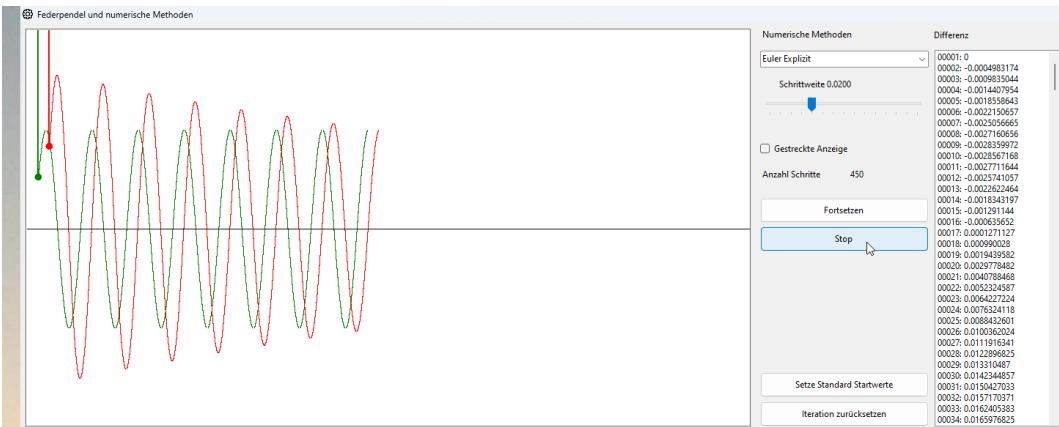
Das rote Pendel schwingt ebenfalls wie ein reales Pendel. Die Differenz rechts ist Null.

9

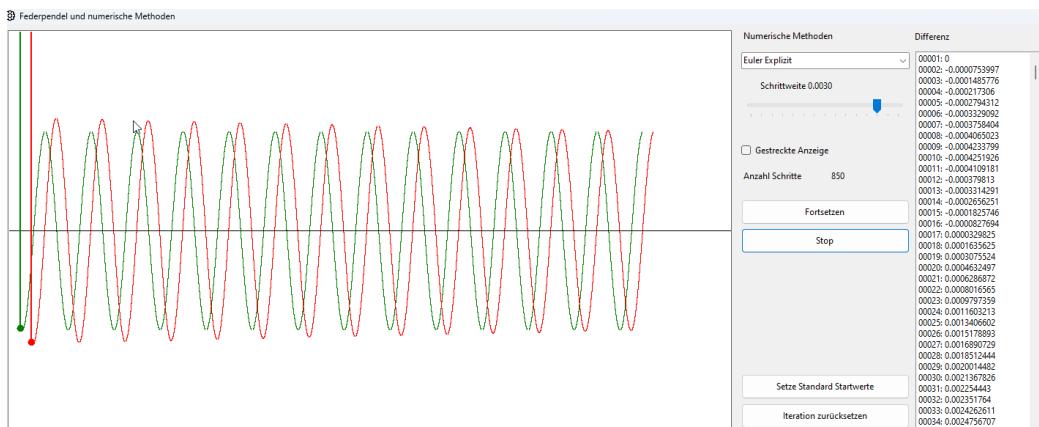
Hier wird die Iteration zurückgesetzt und die Pendel werden in die Standard-Ausgangsposition gebracht.

10

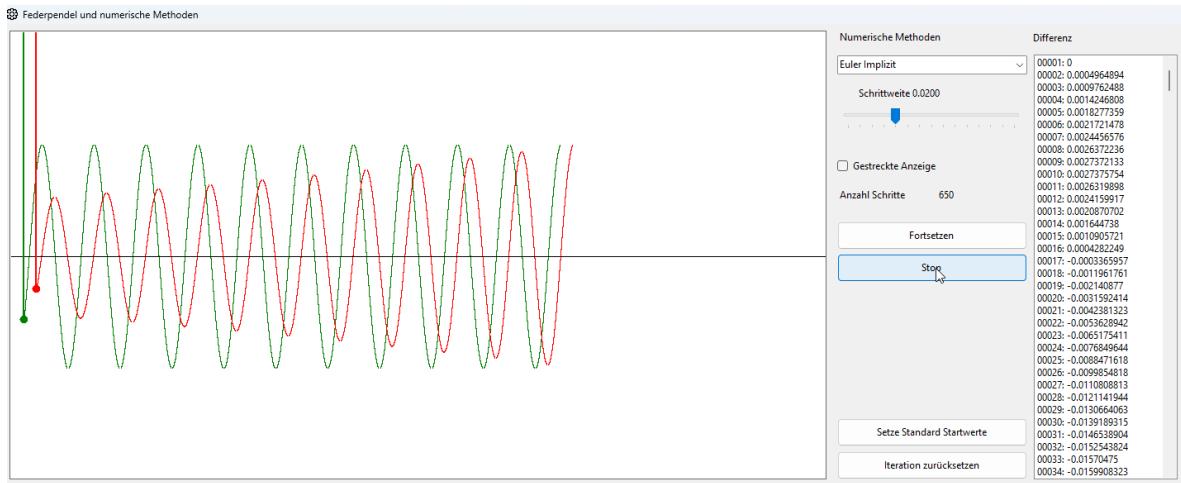
Hier wird nur die Iteration zurückgesetzt.



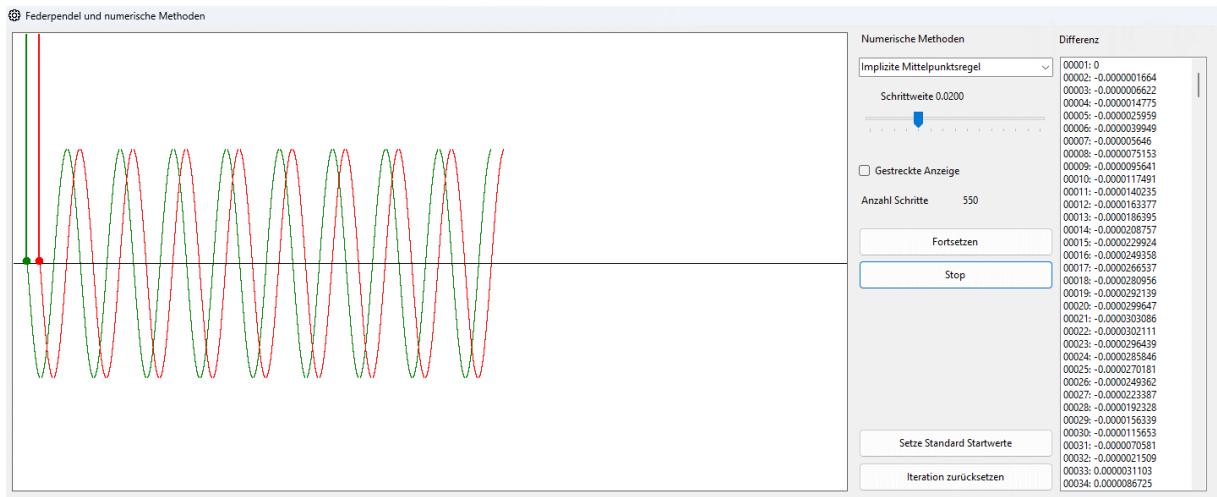
Das rote Pendel wird durch die «Euler Explizit» Methode angenähert (siehe math. Dok.)



Dieselbe Situation, aber mit einer kleineren Schrittweite für das rote Pendel:
Es dauert länger, bis die Bahnen voneinander abweichen

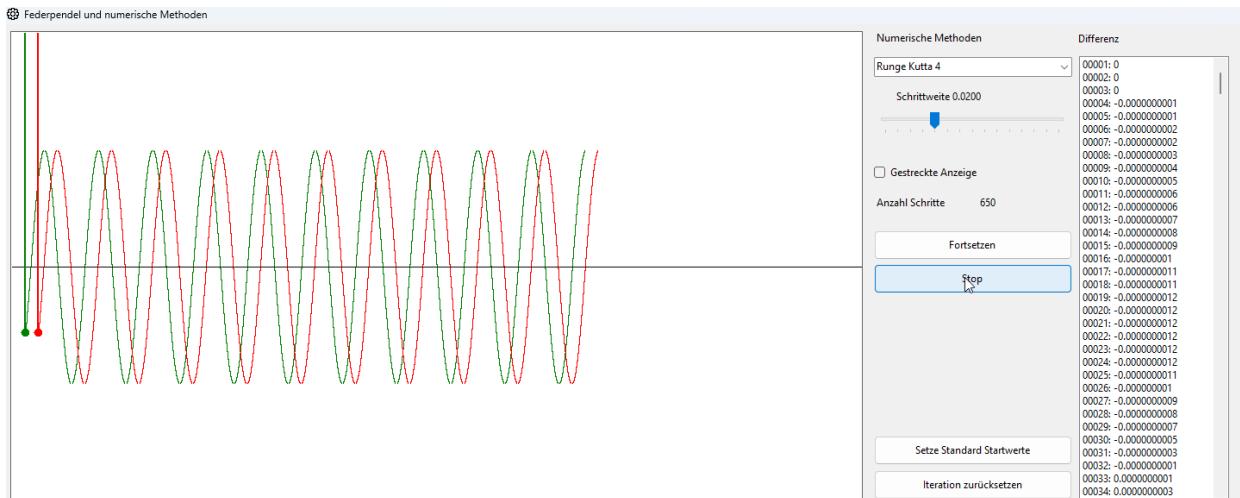


Hier schwingt das rote Pendel angenähert mit der «Euler Implizit» Methode



Die «implizite Mittelpunktregel» liefert eine gute Annäherung für das rote Pendel.

Man sieht aber rechts, dass die Differenz der Positionen nicht Null ist



Am besten schneidet das «Runge Kutta» Verfahren 4. Ordnung ab.

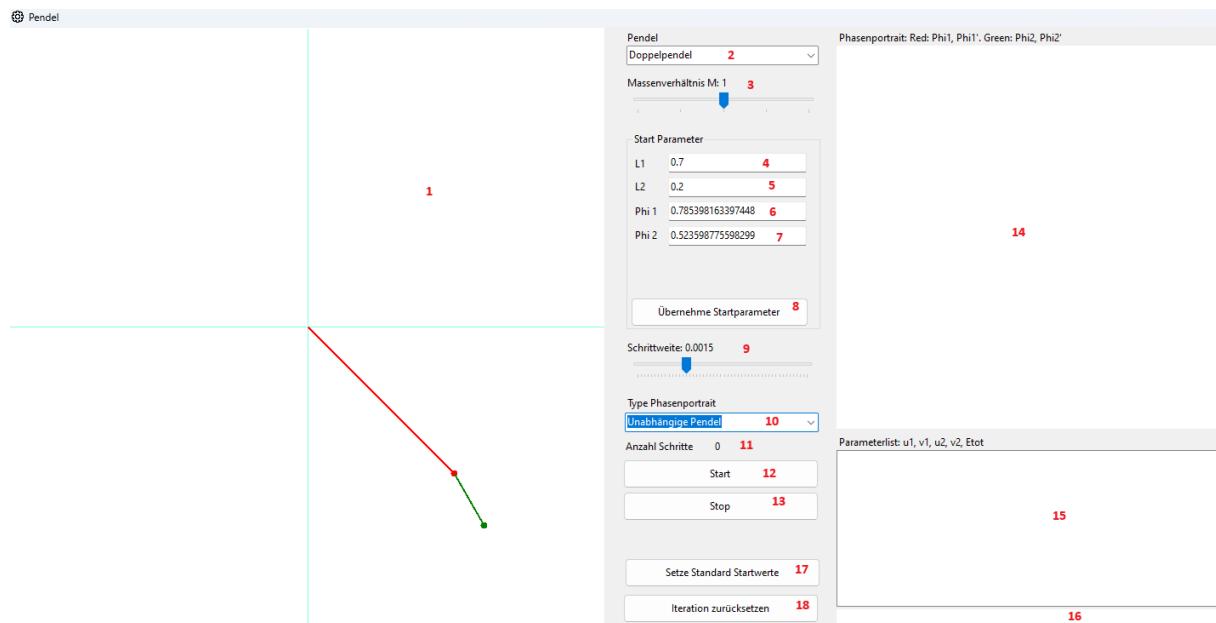
Siehe die kleinen Differenzen rechts.

6.2. Pendel

Hier werden folgende gekoppelte Pendel zur Untersuchung angeboten:

- Doppelpendel
- Schwingendes Federpendel
- Horizontales Rüttelpendel

Durch das Menü «Mechanik – Pendel» wird folgendes Fenster geöffnet:



Fenster zur Untersuchung der verschiedenen Pendel – hier das Doppelpendel

1

Hier wird die Bewegung des Pendels aufgezeigt.

2

Auswahl des Pendels.

3

Je nach Pendel kann hier ein weiterer Parameter festgelegt werden.

4, 5, 6, 7

Hier werden die spezifischen Startparameter angezeigt oder können eingegeben werden. Je nach Pendel können das bis zu 6 verschiedene Parameter sein.

8

Durch diesen Knopf können eingegebene Startparameter dem Pendel übergeben werden.

9

Schrittweite des Runge Kutta Verfahrens.

10

Auswahl des Typs des Phasenportraits. Zur Auswahl stehen:

- Beide Pendel unabhängig voneinander
- Torus oder Zylinder (je nach Pendel)
- Poincaré Schnitt

11

Anzeige der Anzahl Schritte der Iteration.

12

Start der Iteration.

13

Stopp der Iteration.

14

Anzeige des Phasenportraits.

15

Protokoll der relevanten Parameter und Anzeige der aktuellen Energie relativ zur Gesamtenergie am Start.

16

Diese Pendel reagieren sehr sensiv auf kleine Änderungen des Startwertes und damit auch auf kleine Fehler bei der Iteration. Mit Sicherheit ist die approximierte Bewegung nach wenigen Schritten nicht mehr realistisch. Um das mindestens teilweise zu kontrollieren, wird in diesem Balken die Gesamtenergie dargestellt. Diese sollte konstant bleiben. Siehe weiter unten.

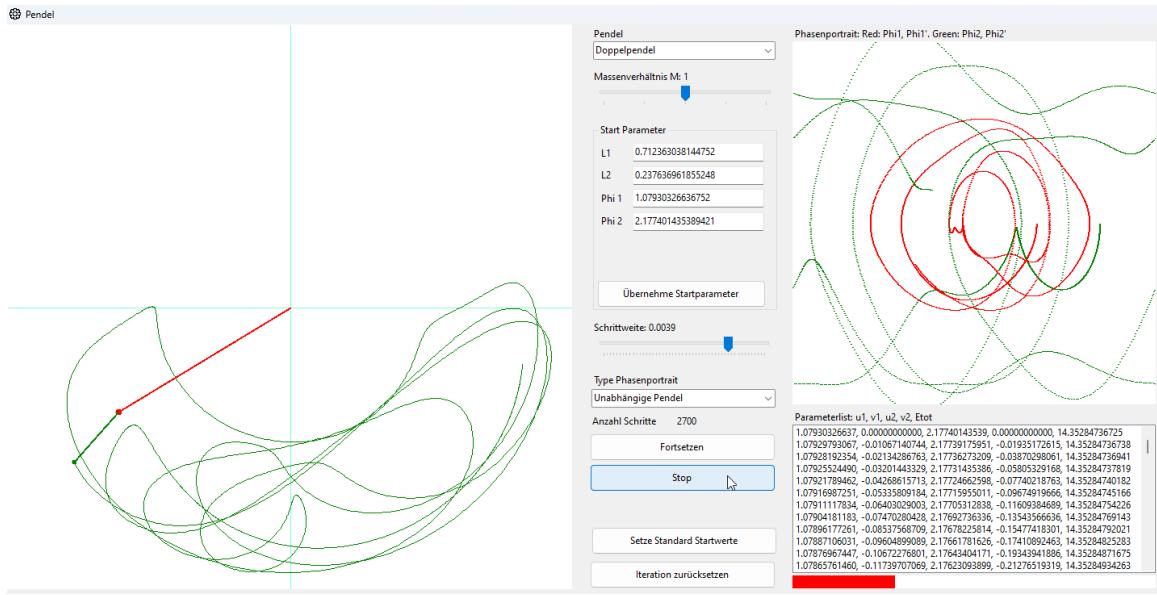
17

Hier wird die Iteration zurückgesetzt und die Startparameter werden auf Standard gesetzt.

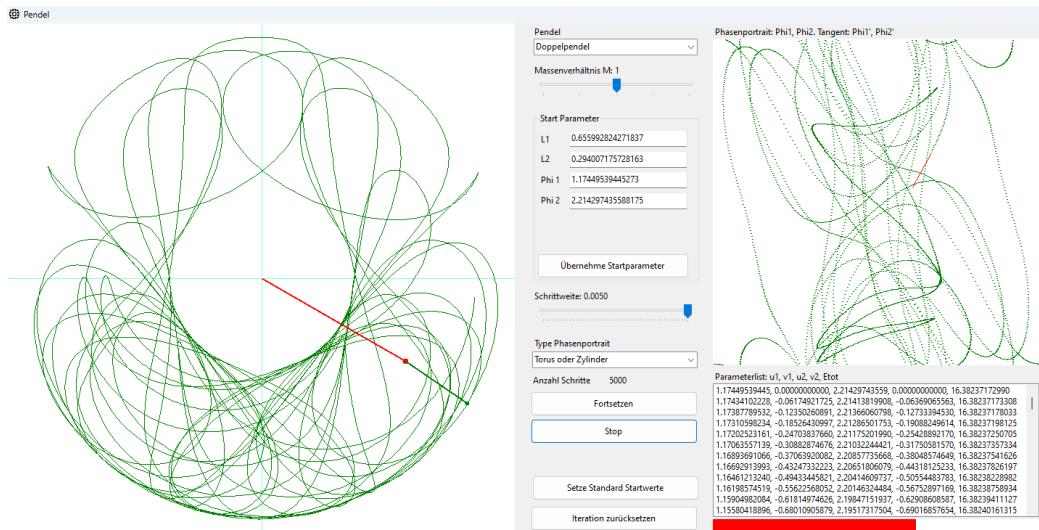
18

Hier wird nur die Iteration zurückgesetzt.

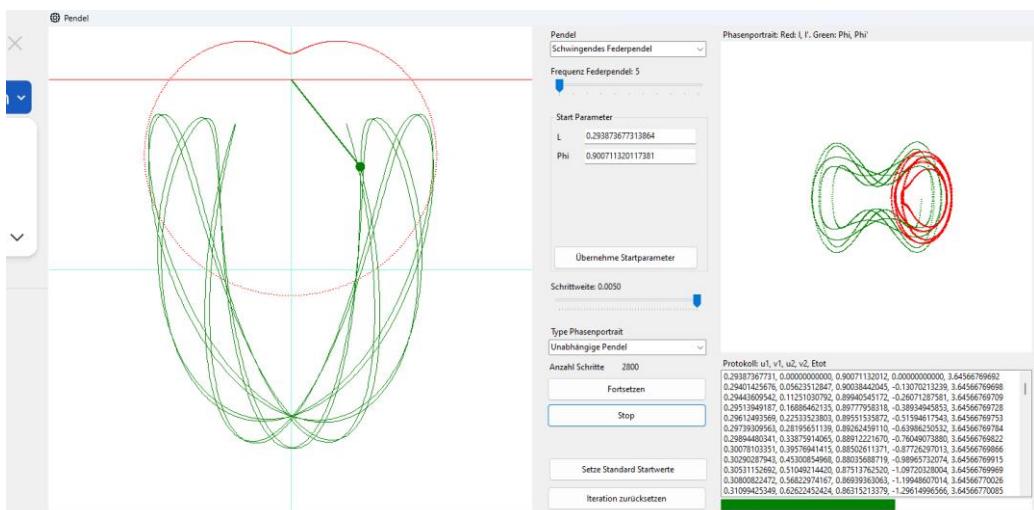
Das Runge Kutta Verfahren (und auch bessere Verfahren) zeigen die «echte» Bewegung des gekoppelten Pendels nur über wenige Schritte. Anschliessend reagiert das Pendel zu sensitiv auf die Ungenauigkeiten der Iteration. Deshalb wird durch die Anzeige der aktuellen Energie darauf hingewiesen, wenn diese von der Gesamtenergie am Start mehr als $\pm 10\%$ abweicht. Liegt die Abweichung unter diesem Wert, erscheint der entsprechende Balken der Energieanzeige grün. Ist die aktuelle Energie zu tief, dann erscheint der Balken violett. Ist sie zu hoch, erscheint er rot.



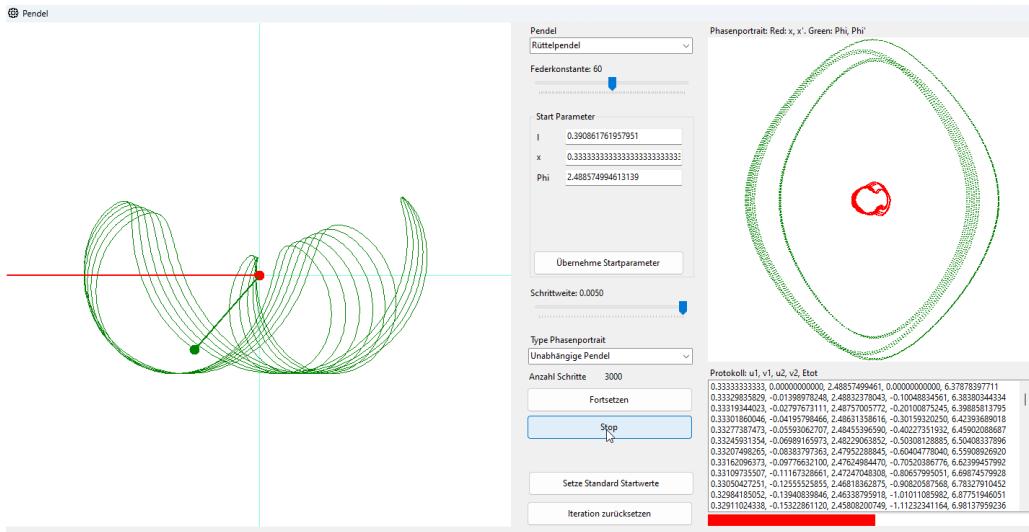
Bewegung des Doppelpendels: Die Energie ist momentan zu hoch.



Darstellung der Bewegung auf dem Torus



Bewegung des schwingenden Federpendels



Bewegung des Rüttelpendels