

Newton Fraktale

Yaroslav Nalivayko

Betreuer: M.Sc. Benjamin Maier

Zusammenfassung Newton Fraktale sind die Teilmenge der mathematischen Fraktale, die durch die Einsetzung des Newton Verfahrens für die Lösung der nichtlinearen Gleichungen auf der komplexen Ebene erscheinen.

1 Einleitung

Newton Fraktale stellen eine interessante Klasse der mathematischen Fraktalen dar. Im Rahmen dieser Arbeit werden Theoretische Grundlagen vorgestellt und ein Programm für die Visualisierung der Fraktale entwickelt. In der letzten Sektion findet die Analyse mancher interessanten Funktionen statt.

2 Theoretische Grundlagen

Hier werden wichtigste Grundbegriffe erläutert.

2.1 Numerische Mathematik

Die numerische Mathematik beschäftigt sich als Teilgebiet der Mathematik mit der Konstruktion und Analyse von Algorithmen für kontinuierliche mathematische Probleme. [Tre92] Sie wird oft benutzt, um approximative Lösungen mit Hilfe des Rechners zu finden.

2.2 Newton Verfahren

Newton Verfahren ist das iterative numerische Verfahren, das die Wurzel gegebener Funktion findet. Die Methode ist nach Sir Isaac Newton benannt.

Wir interessieren uns in stetig differenzierbaren Funktionen mit nur einer Variable.

$$f(x) = 0$$

Man wählt den Startwert x_0 manuell und benutzt die iterative Methode, soweit eine akzeptable Lösung gefunden wird.

$$x_{n+1} = x_n + \frac{f(x_n)}{f'(x_n)}$$

Gewöhnlich wählt man eine zulässige Abweichung ε und eine maximale Anzahl der Schritte N . Nach jedem Schritt der iterative Methode prüft man, ob $f(x_n) < \varepsilon$ ist. In

diesem Fall ist die Lösung gefunden. Und falls $n > N$, dann ist die Lösung unauffindbar in die akzeptable Anzahl der Schritte. Iterative Prozess der Annäherung von x_n zu $x_n + 1$ heißt die Konvergenz, und falls x_0 zu x_n kommt, heißt das, dass x_0 gegen x_n konvergiert.

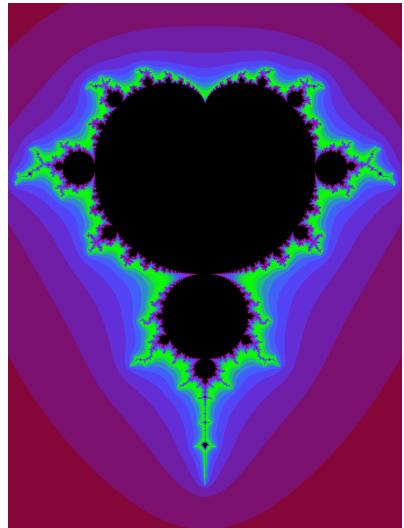
2.3 Fraktale

Fraktal (lateinisch *fractus* - gebrochen) ist ein von Mathematiker Benoît Mandelbrot geprägter Begriff, der bestimmte natürliche oder künstliche Gebilde oder Muster bezeichnet. Diese Gebilde oder Muster weisen einen hohen Grad von Skaleninvarianz bzw. Selbstähnlichkeit. Das ist beispielsweise der Fall, wenn ein Objekt aus mehreren verkleinerten Kopien seiner selbst besteht. [fra]

Die Figur 1a stellt ein Beispiel für ein künstliches Fraktal, und die Figur 1b für ein mathematisches Fraktal.



(a) Romanesco. Ein natürliches Fraktal. [Mak]



(b) Mandelbrot. Ein künstliches Fraktal. [Bey]

2.4 Newton Fraktale

Diese Fraktale erscheinen sich, wenn man das Newton Verfahren für Auffinden der Wurzeln der nichtlinearen Gleichungen auf Komplexe Ebene einsetzt. Genauer gesagt, soll man versuchen, die Wurzel für jeden Punkt des gesuchten Bildes durch Newton Verfahren zu finden.

Zum Beispiel nehmen wir die Funktion $f(x) = x^3 - 1$. Diese Funktion hat drei Lösungen auf Komplexe Ebene: $1, -\sqrt[3]{-1}$ und $(-1)^{2/3}$. Die Näherungswerte in Koordinatenform sind $(1, 0)$, $(-0.5, 0.866)$ und $(-0.5, -0.866)$. Die Punkte des Bildes, die sich durch das Newton Verfahren zu entsprechenden Wurzeln annähern, werden entsprechend mit rot, blau und grün gefärbt. Das Fraktal auf dem Bild 2 repräsentiert die gewählte Umgebung. Die Ursachen der komischen Mischung werden in der Sektion 4 erläutert.

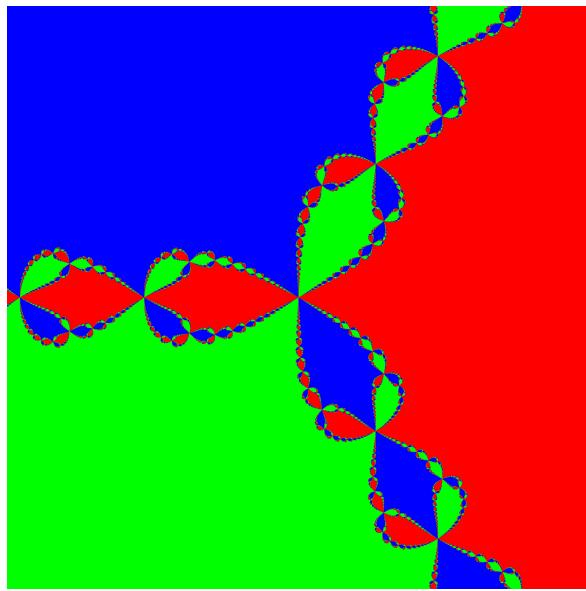


Abbildung 2: Newton Fraktal für $f(x) = x^3 - 1$

```
BufferedImage bi = new BufferedImage(width, height, TYPE_INT_ARGB) ;  
ImageIO.write(bi, "png", outputfilename);
```

Listing 1: Ein Beispiel für die PNG Generierung

3 Visualisierung

In diesem Abschnitt wird ein Programm entwickelt, das Bilder mit der Newton Fraktale generiert. Außerdem wird es ein Beispiel vorgestellt, wie man Animationen erzeugen kann.

3.1 Bildgenerierung

Hier wird ein Programm beschreibt, die Newton Fraktale visualisiert. Als Programmiersprache wurde Java gewählt.

Bibliotheken Zusätzlich zu allgemeine Java-Bibliothek wurde noch das *javax.imageio.ImageIO* verwendet. Das ist die öffentliche Bibliothek für die Arbeit mit Bildern, die, unter anderem, für das *.png* Format geeignet ist. Das PNG (Portable Network Graphics) Format passt für unsere Ziele ideal, wegen kleines Gewicht und keinen Qualitätsverlust der Dateien.

Bild Mit Hilfe der *javax.imageio.ImageIO* kann man sehr einfach die *.png* Bildern erzeugen. Ein Beispiel ist in der Auflistung 1. Um den Bildinhalt einzusetzen, benutzt man den *bi.setRGB(x, y, Color)*; Befehl.

Natürlich soll man einen bestimmten Bereich des Fraktales wählen, um es zu zeichnen. Bei dem Programm wählt man den Startpunkt durch X und Y Koordinatenwerte, die Bildschirmgröße $Size$ und die Auflösung $resolution$.

Komplexe Zahlen Für die Bearbeitung der komplexen Zahlen ist eine neue Klasse implementiert, die die Arbeit mit komplexen Zahlen erleichtert. So kann man komplexe Zahlen addieren, abrechnen, multiplizieren, dividieren und potenzieren.

Newton Verfahren Um Newton Verfahren zu benutzen, soll man manuell die Ableitung berechnen, die iterative Newton Funktion ausfinden und im Programmcode umwandeln. Außerdem assoziiert man mit jeder Wurzel eine Farbe, um später das Bild zu zeichnen. Als Beispiel nehmen wir die Funktion $f(z) = z^3 - 2z + 2$. Die entsprechende Ableitung hat die Form $f'(z) = 3z^2 - 2$ und die iterative Funktion des Newton Verfahrens $z_{n+1} = \frac{2z_n^3 - 2}{3z_n^2 - 2}$. Und hier steht der Programmcode für die iterative Funktion.

```
devide(mult(hoch(wert, 3), 2).add(-2), mult(hoch(wert, 2), 3).add(-2));
```

Die Funktion wird iterativ für jedes Pixel des Bildes verwendet, um die Farbe zu berechnen. Natürlich, man soll nach jedem Schritt Prüfen, ob die maximale Anzahl der Schritte N nicht überschreitet wird und ob der Punkt schon die Wurzel ist. Das heißt, dass es für jeden gefundenen Punkt z_n geprüft werden soll, ob $n > N$ oder $|f(z_n)| < \varepsilon$ ist. Falls einer der beiden Fälle eintritt, bricht man die iterative Berechnung ab. Im ersten Fall bezeichnet man das Punkt mit weiße Farbe, im zweiten - mit Farbe assoziierte mit der gefundenen Wurzel.

Lösungen Da man manuell Farben für unterschiedliche Wurzeln wählen will, soll man alle Wurzeln mit assoziierten Farben im Programmcode eintragen. Ein Beispiel für die Funktion $z^3 - 1$ findet man in der Auflistung 2. Jede Wurzel wird durch zwei Zahlen festgestellt, die reale und imaginäre Parts abbilden. Letzte drei Argumente definieren die Farbe der Wurzel in das RGB Format.

Listing 2: Wurzeln mit Farben für $z^3 - 1$

```
points.add(new point(1, 0, 255, 0, 0));
points.add(new point(-0.5, 0.866, 0, 255, 0));
points.add(new point(-0.5, -0.866, 0, 0, 255));
```

Schatten Für bessere Informationswiedergabe darf man optional die Schatten einschalten. Dann je mehr Schritten des Newton Verfahren nötig sind, desto dunkler das Pixel wird.

Programmkode Vollständige Programmkode kann man unter [Nal] finden.

3.2 Animation

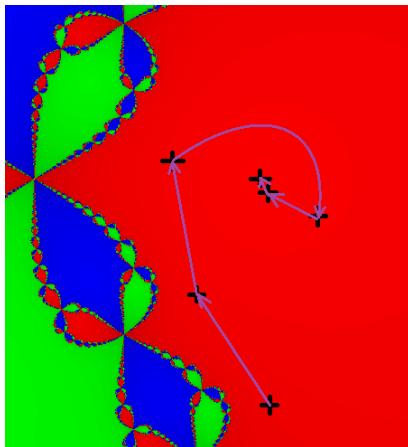
Einer der einfachsten Wegen für die Animationserzeugung wurde gewählt. Das Programm wurde so geändert, dass *size* und *outputfilename* variabel werden. Dann im Zyklus wird eine Reihenfolge der Bildern generiert, so dass das neue Bild die Annäherung des Alten wird. Die Animation wird aus dieser Reihenfolge der Bildern mit Hilfe des [ani] generiert.

4 Analyse

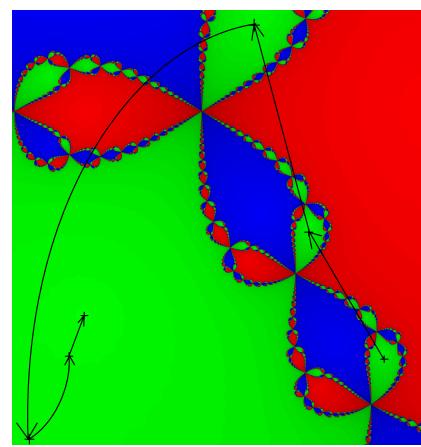
Manche Newton Fraktale werden vorgestellt und analysiert.

4.1 $z^3 - 1 = 0$

Wie oben schon gesagt war, hat diese Gleichung drei Lösungen auf Komplexe Ebene: $+1, -\sqrt[3]{-1}$ und $(-1)^{2/3}$. Entsprechende Newton Fraktal ist auf dem Bild 2 visualisiert.



(a) Die Konvergenz des Punktes $(1, 1)$ für
 $f(z) = z^3 - 1$



(b) Die Konvergenz des Punktes $(0.9, 1)$ für
 $f(z) = z^3 - 1$

Wenn man versucht die Grenze zwischen zwei Zonen zu annähern, sieht man, dass dort immer dritte Zone vorkommt. Das wiederholt sich rekursiv bei Annäherung. In anderen Worten, wenn man einen Anfangspunkts z_0 wählt, der gegen $+1$ konvergiert und einen Punkt z_1 , der gegen $-\sqrt[3]{-1}$ konvergiert, dann existiert es zwischen z_0 und z_1 immer einen dritten Punkt z_2 , der noch näher zu z_0 als z_1 liegt und zu $(-1)^{2/3}$ konvergiert.

Wichtigste Frage ist: warum sehen die Zonen nicht wie die einfachen 120-Grad-Sektoren aus? Anfang letztes Jahrhunderts gelang es zwei französischen Mathematiker Gaston Julia und Pierre Fatou zu zeigen, dass die Grenzpunkten eines Einzugsgebiets die Grenzpunkte aller Einzugsgebiete sind. Folglich können Iterationen mit mehr als zwei Einzugsgebieten keine einfach zusammenhängenden Liniensegmente als Gebietsgrenzen in 2D haben. Solche Grenzen müssen zwangsläufig fraktalen Natur sein, bestehend

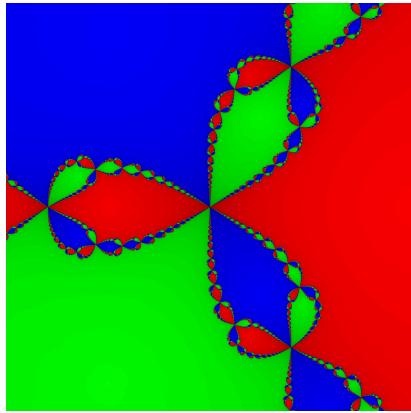
aus vollständig separaten Punktmengen - sozusagen eine unendlich feine Staubwolke aus nichtabzählbar vielen Staubpartikelchen. [Sch94] Machen wir uns damit ein bisschen vertraut, indem wir die Nullstelle (der Punkt $0 : 0$) anschauen. Logischerweise soll die Nullstelle zwischen allen Zonen liegen. Die Null selbst konvergiert überhaupt nicht, da die Einsetzung $z = 0$ zu Division durch Null bei Newton Verfahren folgt. Aber diese Nullstelle ist nicht allein, außer es existieren noch unendlich viele Punkte, die gegen die Nullstelle konvergieren und für die auch gilt, dass daneben alle Zonen vorhanden sind. (Das folgt aus der lineare Natur des Newton Verfahrens.) Diese Punkten, die gegen die Null konvergieren, bilden diese komische Grenze Zwischen den Zonen.

Zum Beispiel betrachten wir die Punkte $(1, 1)$ und $(0.9, 1)$. Die Konvergenz des Punktes $(1, 1)$ wird auf dem Bild 3a vorgestellt.

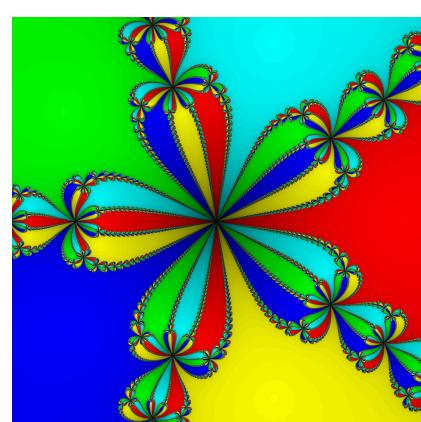
Der Punkt liegt weit genug von der Grenze und wird eindeutig konvergiert, trotzdem sieht man so ein Sprung, die sehr oft bei Newton Verfahren eintreten können.

Jetzt betrachten wir das Punkt $(0.9, 1)$ auf dem Bild 3b. Die Konvergenz des Punktes zeigt eindeutig, wie die iterative Schritte des Newton Verfahrens funktionieren und wie man zu solchen Bildern kommt. Erste zwei iterative Schritte behalten sich gleich wie bei dem Punkt $(1, 1)$, was die lineare Natur des Newton Verfahrens bekräftigt. Aber dann, neben der Nullstelle, wird es nach grünen Gebiet geworfen. Die Punkten neben Grenzen bewegen sich zuerst in die Richtung der Nullstelle und dann rasant in die passende Zone springen.

Zusätzliches Beispiel wird auf dem Bild 4a vorgestellt, wo die helle Bereiche die am schnellsten konvergierende Punkte bezeichnen.



(a) Die Konvergenzgeschwindigkeit durch die Helligkeit für $f(z) = z^3 - 1$



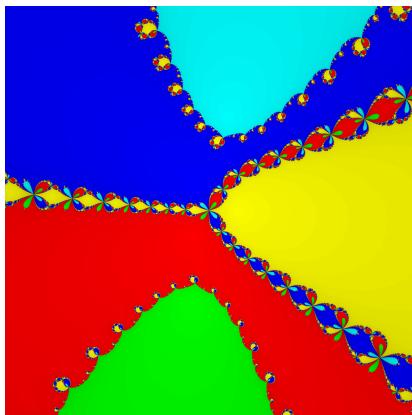
(b) Das Newton Fraktal für $f(z) = z^5 - 1$

4.2 $z^5 - 1 = 0$

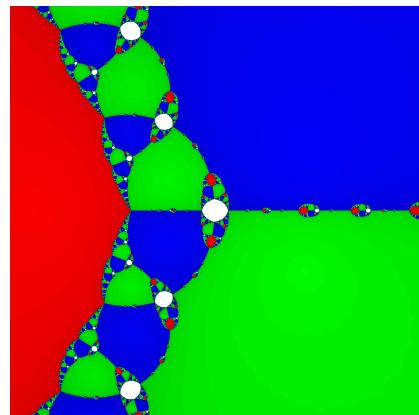
Das Bild 4b illustriert das entsprechende Newton Fraktal. Der einziger Unterschied liegt in der Anzahl der Zonen und Lösungen. Wie bei $z^3 - 1 = 0$ sind die Grenzpunkten eines Einzugsgebiets die Grenzpunkte aller Einzugsgebiete, was für die fünf Zonen interessanter aussieht.

4.3 $z^5 + (5 + 2i)z^3 - 2 - i$

Das entsprechende Newton Fraktal (das Bild 5a) besitzt eine interessante Struktur. Der Grund dazu liegt in der Position der Wurzeln. Die Nullstelle, die grüne und die rote Zonen liegen fast auf die Gerade. Das Gleiche gilt für die Nullstelle, die blaue und die hellblaue Zonen. Unabhängig davon, dass die hellblaue und die grüne Zonen weit weg von der Nullstelle liegen und mit der roten und blauen Zonen abgegrenzt sind, konvergieren manche Punkte neben der Nullstelle gegen das Grün und das Hellblau.



(a) Das Newton Fraktal für $f(z) = z^5 + (5 + 2i)z^3 - 2 - i$



(b) Das Newton Fraktal für $f(z) = z^3 - 2z + 2$

4.4 $z^3 - 2z + 2$

Das Bild 5b illustriert das Newton Fraktal für diese Funktion. Wie man sieht, hier gibt es ziemlich große weiße Bereiche. Was passiert dort? Um diese Frage zu beantworten, sollen wir ein paar Schritte des Newton Verfahrens manuell ausführen. Die iterative Funktion hat die folgende Form:

$$z_{n+1} = \frac{2z_n^3 - 2}{3z_n^2 - 2}$$

Probieren wir das Punkt $(0,0)$, oder $z_0 = 0$ ist. Nach der Einsetzung in die iterative Formel bekommt man $z_1 = 1$. Sieht noch gut aus, aber was passiert weiter? Erstaunlicherweise ist z_2 wieder gleich 0. So springt der Wert zwischen 0 und 1, was ein gutes Beispiel ist, warum man die Anzahl der Schritten begrenzen soll. Aber was passiert neben dem Punkt $(0,0)$, sodass diese weißen Gebiete so groß sind? Iterieren wir der Punkt $(0.1, 0)$, der ziemlich nah zu der Nullstelle liegt. Dann ist $z_0 = 0.1$; $z_1 = 1.014213$ und $z_2 = 0.079655$. Man sieht, dass z_2 noch näher zu der Nullstelle liegt, als z_0 . Je mehr iterative Schritte man ausführt, desto näher es zu der Nullstelle hintritt. Wenn man andere Punkte aus der Umgebung der Nullstelle zu iterieren versucht, werden die Punkten, die nah genug liegen, gegen die Nullstelle konvergieren. So werden die weißen Fällen gebildet, die diese Funktion auszeichnen.

5 Zusammenfassung

Numerische Lösungen von nichtlinearen Gleichungen durch Approximation sind in der modernen Welt überall gebraucht. Die approximative Berechnungen bringen viele unerwartete Probleme und interessante Ereignisse. Newton Fraktale ist eine der Ereignisse. Die Erforschung des Phänomens ist wichtig für unsere Verständnis des Newton Verfahrens.

Im Rahmen dieser Arbeit sind theoretische Grundlagen vorgestellt, wichtigste Begriffe erläutert und erklärt. Dann hat man die Software entwickelt, die Newton Fraktale visualisiert. Es ist eine Methode vorgestellt, die beschreibt, wie man Animationen erzeugen kann. Danach sind manche Newton Fraktale analysiert und die Ursachen für ihre interessante Struktur vorgestellt.

Literatur

- ani. <https://toolson.net/GifAnimation/Create>
- Bey. https://de.wikipedia.org/wiki/Datei:Mandelbrot_set_with_coloured_environment.png
- fra. *Fraktal.* <https://de.wikipedia.org/wiki/Fraktal>. – Online-Ressource
- Mak. <https://de.wikipedia.org/wiki/Datei:Romanesco.jpg>
- Nal. NALIVAYKO, Yaroslav: *Newton Fraktale Visualisierung.* https://github.com/Jery77/Simulation_Prog/tree/master/src, . – Accessed Juni 4, 2017
- Sch94. SCHROEDER, Manfred ; SCHLIERF, Markus (Hrsg.): *Fraktale, Chaos und Selbstähnlichkeit.* Spektrum, 1994
- Tre92. TREFETHEN, Lloyd: The definition of numerical analysis. In: *SIAM News* (1992)