

Anleitung: GeZA für Android

GeZA ist eine Android App mit der 1-dimensionale zelluläre Automaten untersucht und in attraktiven Grafiken dargestellt werden können. GeZA ist kostenlos, werbefrei und über den Google Play Store installiert werden.

Inhaltsverzeichnis:

Bedienung.....	2
allgemein.....	2
Regel-Editor.....	4
Farben setzen.....	5
Datei speichern.....	6
Datei öffnen.....	7
Grundlagen.....	8
"totalistische" zelluläre Automaten.....	8
einfache Wolfram Automaten.....	9
NaSch-Modell.....	10
TwoSteps.....	10
Zhabotinsky.....	11
Block Cellular Automaton.....	12
Optionen.....	13
totalistisch.....	13
Wolfram.....	14
NaSch-Modell.....	15
TwoSteps.....	16
Zhabotinsky.....	17
Optionen Block CA.....	18
Credits.....	19
Links.....	20






Bedienung

allgemein

Nach Start von GeZA erscheint die Darstellung eines Zellularautomaten. Durch Wischgesten (siehe Tabelle unten) können sie diverse Funktionen auslösen:

Wisch-Geste	Funktion
kurzer Touch	Start bzw. Stopp der Animation
unten nach oben	die nächsten Generationen anzeigen
oben nach unten	Start-Generationen anzeigen, Wurde diese schon angezeigt, so wird der Standard-Startzustand (1 Pixel mit Zustand 1) angezeigt.
rechts nach links langsam	neue zufällige, symmetrische Start-Generation anzeigen
rechts nach links schnell	Anzeige um halbe Bildschirmbreite nach links verschieben
links nach rechts langsam	neue zufällige, nicht symmetrische Start-Generation anzeigen.
links nach rechts schnell	Anzeige um halbe Bildschirmbreite nach rechts verschieben

Neben den Wischgesten stehen weitere Funktionen über ein Drop-Down-Menü und Icons in der Titelzeile zur Verfügung.

Menü-Eintrag	Icon	Funktion
zoom in		Ansicht heranzoomen für mehr Details.
zoom out		Ansicht verkleinern für größere Übersicht.
Beispiele		Zeigt die Liste der vorhandenen Beispiele an. Durch Touch auf eines der Icons wird der zugehörige Automat direkt gestartet.
Optionen		es erscheint ein Dialogfenster, in dem kontextbezogenen Einstellungen vorgenommen werden können.
Farben setzen		Farbvorgaben für die verschiedenen Zellenzustände verändern.
Anleitung	-	Beschreibung ansehen
als Hintergrund	-	das aktuelle Bild kann als Hintergrundbild (wallpaper) festgelegt werden.
Totalistisch	-	auf den totalistischen Modus einstellen. siehe dazu Grundlagen totalistisch .
Wolfram	-	auf den Wolfram-Modus einstellen. siehe dazu Grundlagen Wolfram .
NaSch-Modell	-	auf das NaSch-Modell einstellen. siehe dazu Grundlagen NaSch-Modell .

Regel-Editor

Beim Klicken in den Optionen auf die Regel öffnet sich der Regel-Editor.



Oben ist der aktuelle Regel-String zu sehen. Durch Touch auf diesen String können Sie den Cursor setzen. Die Zahl in rot darunter zeigt die aktuelle Cursor-/Stringposition an. Dies ist hilfreich, weil das Zeichen den Wert vorschreibt, den eine Zelle annimmt, wenn die Summe der Nachbarwerte gleich der Positionsnummer ist.

Betätigen des Buttons "Anwenden" übergibt die neue Regel an den Optionen-Dialog und geht dorthin zurück.

Steuerung:

Button	Funktion
<---	Cursor um 1 Position nach links.
--->	Cursor um 1 Position nach rechts.
Hilfe	ruft diese Hilfeseite auf.
< del	löscht das Zeichen links von Cursor.
del >	löscht das Zeichen rechts vom Cursor.

Bedeutung der Regelzeichen:

Zeichen	Bedeutung
=	Die Zelle behält ihren Wert.
+	Der Wert der Zelle wird um 1 erhöht, aus 9 wird Null.
-	Der Wert der Zelle wird um 1 erniedrigt, Null bleibt Null.
!	Aus Null wird 1, ansonsten erhält die Zelle den Wert Null.
?	Die Zelle erhält einen zufälligen Wert.
>	Die Zelle erhält den größten Nachbarzellenwert
<	Die Zelle erhält den kleinsten Nachbarzellenwert.

Bei sehr langen Regel-Strings gibt es ein Anzeigeproblem. Es ist ein bekanntes Problem!

Farben setzen

Hier ist die aktuelle Farbauswahl für die Zustände (0-9) der Zellen zu sehen.

Abhängig von Kontext (totalistisch, Wolfram, NaSch) werden 10, 2 oder 7 Zellen angezeigt.



Nach Klicken auf eine solche farbige Zelle öffnet sich die Farbauswahl, wo die gewünschte neue Farbe zugeordnet werden kann.

Klicken Sie auf den Button "Anwenden" um die neue Farbpalette zu übernehmen und Sie gelangen zurück zur Hauptansicht.

Mit dem Button "Standard" wird auf die standardmäßige Farbpalette zurückgestellt.

Datei speichern

Hier kann der aktuelle Zellularautomat als GeZA-Datei und/oder als Bilddatei gespeichert werden!



Verzeichnis:
/storage/emulated/0/GeZA

Datei (png):
geza_2021-03-27_134343.png

Datei (GeZA):
geza_2021-03-27_134343.gza

Hilfe speichern

Ganz oben wird das Verzeichnis, in das geschrieben werden soll, angezeigt. Es ist fest vorgegeben, ein Wechsel des Verzeichnis ist noch nicht möglich.

Man hat die Möglichkeit 2 Typen von Dateien zu speichern:

- 1) Eine Grafik des aktuellen Bildschirms im PNG-Format
- 2) den aktuellen Zellularautomaten mit allen Parametern und Einstellungen im GZA-Format, Diese Datei kann nur mit der GeZA-App wieder eingelesen werden, aber problemlos auf anderen Android-Geräten mit der GeZA-App.

Welche Datei-Typen geschrieben werden sollen, kann über die Check-Boxen gesteuert werden. Die Dateinamen werden vorgegeben, zusammengesetzt aus "geza_", Zeitstempel und jeweiliger Endung.

Datei öffnen

Hier können abgespeicherte GeZA-Dateien ausgewählt und geöffnet werden.



Die GeZA-Dateien werden in einem fest vorgegeben Verzeichnis gesucht. Dieses Verzeichnis ("GeZA" auf dem internen Speicher) kann noch nicht frei gewählt werden. Der vollständige Name wird in der ersten Zeile angezeigt.

Darunter sieht man eine Liste der gefundenen Dateien (mit Snapshot und Name). Durch Antippen eines Eintrags werden die zugehörigen Informationen noch einmal unter "gewählte Datei" angezeigt. Mit "Datei öffnen" wird der entsprechende Automat geladen und angezeigt. Durch doppeltes Tippen auf einen Listeneintrag wird der Automat sofort gestartet.

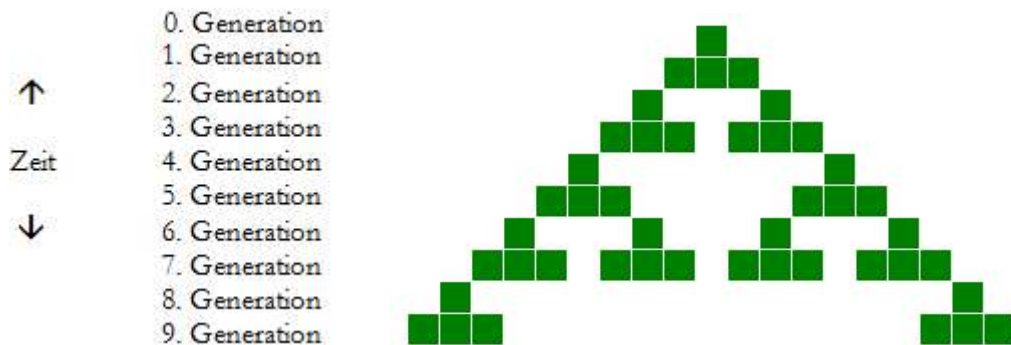
Grundlagen

"totalistische" zelluläre Automaten

Ein zellulärer Automat besteht aus einer Menge von Zellen die jede einen definierten, meistens diskreten, Zustand hat (Anfangszustand). Eine Regel, die den Zustand der Zelle und endlich vieler Nachbarzellen berücksichtigt, bestimmt den Zustand der Zelle in der nächsten Generation. Die Zellen können auf einer Linie (1-dimensional) auf einer Fläche (2-dimensional) oder auch in höheren Dimensionen angeordnet sein.

Einer der bekanntesten zellulären Automaten ist das Game of Life von John Conway aus den 1970er Jahren. Die Zellen sind auf einer Fläche, also 2-dimensional, angeordnet.

GeZa dagegen behandelt ausschließlich 1-dimensionale Zellularautomaten, d.h. die Zellen sind entlang einer Linie aufgereiht. Die einzelnen Generationen werden als einzelne Zeilen untereinander dargestellt. Man kann also auf der Fläche des Bildschirms gleichzeitig die räumliche und zeitliche Entwicklung darstellen. Hier am Beispiel der Regel 0100, bzw. 01 (nicht aufgeführte Stellen werden automatisch mit Nullen aufgefüllt):



Eine Zelle kann nur den Zustand 0 (=weiß) oder 1 (=grün) besitzen.

Als Anfangszustand (=0. Generation) wurde eine Zelle auf 1 (grün), alle andern auf 0 (weiß) gesetzt. Die Regel 01 (3 Zellen-Modus) bedeutet dabei folgende Berechnungsvorschrift für die Bildung der nächsten Generation:

Summe von 3 benachbarten Zellen (links, Mitte, rechts)	0	1	2	3
Zustand der mittleren Zelle in der nächsten Generation	0	1	0	0

Dies ist eine links/rechts-symmetrische Regel. Zellularautomaten deren Regel nur von der Summe der Nachbarzellen abhängen werden **"totalistisch"** genannt.

Die Ziffer an der n-ten Position der Regel gibt den Zustand der neuen Zelle an, wenn die Summe der 3 Nachbarzellen n-1 ist. Ist die Summe beispielsweise 3, so erhält die Zelle in der nächsten Generation den Wert der 4. Stelle in der Regel. Ist die Stelle nicht vorhanden wird Null gesetzt. Diese Schreibweise weicht von der von Stephen Wolfram verwendeten ab.

Sieben zusätzliche Zeichen haben die folgende spezielle Bedeutung:

Zeichen	Bedeutung
=	Die Zelle behält ihren Wert.
+	Der Wert der Zelle wird um 1 erhöht, aus 9 wird Null.
-	Der Wert der Zelle wird um 1 erniedrigt, Null bleibt Null.
!	Aus Null wird 1, ansonsten erhält die Zelle den Wert Null.
?	Die Zelle erhält einen zufälligen Wert.
>	Die Zelle erhält den größten Nachbarzellenwert.
<	Die Zelle erhält den kleinsten Nachbarzellenwert.

einfache Wolfram Automaten

In diesem einfachen Wolfram-Zellularautomaten können die Zellen nur die Werte 0 oder 1 annehmen.

Für die Transformation zur nächsten Generation werden der Zustand der linken Nachbarzelle, der Zelle selbst und der rechten Nachbarzelle betrachtet. Das ergibt $2^3 = 8$ mögliche Nachbarschaftszustände. Jedem dieser 8 Nachbarschaften kann nun eine 0 oder 1 zugeordnet werden. Dies ist dann der Wert der betrachteten Zelle in der nächsten Generation. Diese 8 Nullen oder Einsen kann man als 8stellige binäre Zahl lesen. Das ist dann die eindeutige Transformationsvorschrift (Regel).

Nachbarn	111	110	101	100	011	010	001	000
Wert neue Gen.	0	0	0	1	1	1	1	0

In obigen Beispiel ist die Regel 00011110 binär bzw. im Dezimalsystem 30. Wegen der großen Bedeutung der Regel 30 startet der Wolfram-Typ in dieser App mit dieser Regel.

Eine weitere wichtige Regel ist 110 (dezimal), Für die Regel 110 wurde bewiesen, dass sie eine universelle Turing-Maschine ist.

In dieser App können Sie die Regeln 0 bis 255 untersuchen.

Weitere Informationen sind dem entsprechenden [Wikipedia-Eintrag](#) zu entnehmen.

NaSch-Modell

Das "NaSch" steht hier für Nagel-Schreckenberg-Modell. Es ist eine Straßenverkehrssimulation und wurde 1992 von den beiden Physikern Kai Nagel und Michael Schreckenberg vorgeschlagen.

Es lieferte erstmals eine Erklärung für die Entstehung eines "Staus aus dem Nichts".

Die Zellen repräsentieren im NaSch-Modell eine Straße gleich langer Abschnitte. Diese Zellen können leer sein oder ein Auto mit einer diskreten Geschwindigkeit enthalten. Die Geschwindigkeiten können 6 verschiedene Werte annehmen (von 0 bis 135 in 27 km/h-Schritten). Es wird ein Anfangszustand zufällig vorgegeben. Die Dichte der Fahrzeuge kann unter "Optionen" festgelegt werden (Standard ist 15%). Ein Auto beschleunigt, wenn es zum einen genügend leere Straßenzellen vor sich hat und noch nicht die Maximalgeschwindigkeit erreicht hat. Ein Auto bremst, wenn es zu wenig freie Straßenzellen vor sich hat. Hinzu kommt noch ein unmotiviertes Abbremsen bei einem Teil der Fahrzeuge. Der Anteil der unmotiviert bremsenden Autos legt der "Trödel-Faktor" fest.

Wert	Bedeutung
0	Straßenzelle ohne Auto
1	Auto stehend (0 km/h)
2	Auto mit Tempo 27 km/h
3	Auto mit Tempo 54 km/h
4	Auto mit Tempo 81 km/h
5	Auto mit Tempo 108 km/h
6	Auto mit Tempo 135 km/h

Fahrtrichtung ist von links nach rechts, die zeitliche Abfolge verläuft von oben nach unten. Man kann sehen, dass Staus sich nach links bewegen, obwohl die Autos nach rechts fahren.

Weitere Informationen sind dem entsprechenden [Wikipedia-Eintrag](#) zu entnehmen.

TwoSteps

Dieser Typ von Zellularautomaten funktioniert wie die "totalistischen". Bei "TwoSteps" werden jedoch abwechselnd unterschiedliche Regeln/Nachbarschaften verwendet.

Also bei einer ungeraden Generation eine andere Regel als bei einer geraden. (Sind beide Regeln und Nachbarschaft gleich, so ist dieser Automat mit dem "totalistischen" identisch!)

Zhabotinsky

Der Name dieses Zellularautomaten geht zurück auf einen chemischen Oszillator der [Belousov-Zhabotinsky-Reaktion](#),

Zur Simulation dieser chemischen Reaktion habe ich die "Misch-Masch-Maschine" von Martin Gerhardt und Heike Schuster verwendet.

In GeZA ist der Algorithmus auf 1-dimensionale Verhältnisse angepasst. Die originale 2-dim-Version finden Sie in der App GeZA-2D!

Zellen können Werte zwischen 0 (gesund), 1 bis 255 (infiziert) und 256 (krank) annehmen.

Wenn eine Zelle den Wert 0 hat (gesund) wird ihr für die neuen Generation der Wert $\lfloor [K/k1] + [I/k2] \rfloor$ zugewiesen.

wobei K die Anzahl der kranken Nachbarn und I die Anzahl der infizierten Nachbarn ist.

k1 ist die Krank-Schwelle (1-5), k2 ist die Infekt-Schwelle (1-5).

Die eckigen Klammern $\lfloor \rfloor$ sagen, dass der Wert abgerundet wird.

Infizierte Zellen (Wert 1-255) wird für die neue Generation $\lfloor [S/A] \rfloor + g$, maximal 256, zugewiesen. S ist die Summe der Werte aller Nachbarzellen und der Zelle selbst. A ist die Anzahl der Zellen. $\lfloor [S/A] \rfloor$ ist praktisch der Mittelwert.

g ist ein konstanter Faktor (1-120) der immer hinzu addiert wird, so etwas wie eine Anregung.

Eine kranke Zelle (Wert = 256) ist in der neuen Generation gesund (Wert = 0).

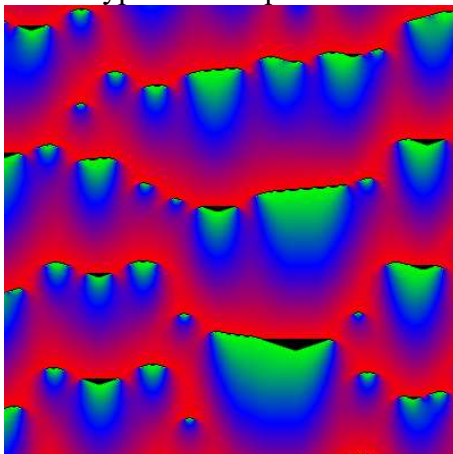
Bei diesem Automaten wird die Nachbarschaft mit $r=3$  verwendet.

Mit anderen Worten kann man den Ablauf so beschreiben:

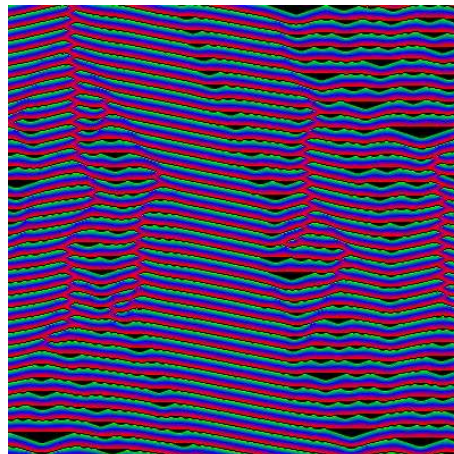
- Gesunde Zellen werden durch infizierte und kranke Nachbarzellen infiziert.
- auf infizierte Zellen wirkt ein Mechanismus aus Anregung und Diffusion bis diese "krank" wird.
- "kranke" Zellen werden sofort wieder gesund und es geht von vorne los.

Es können eine Vielzahl von Mustern, abhängig von Parametern k1, k2, und g, sowie den Anfangsbedingungen, entstehen.

Hier 2 typische Beispiele:



k1=1, k2=2, g=2

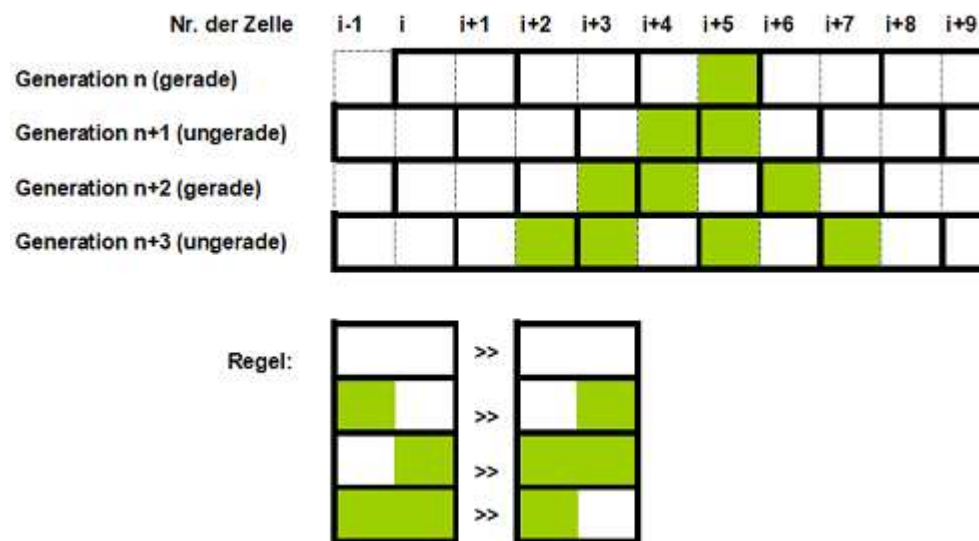


k1=1, k2=2, g=30

Block Cellular Automaton

Bei diesem Typ von zellulären Automaten werden immer mehrere Zellen zu einem Block zusammengefasst. Für diese Blöcke gibt es eine Transformationsregel, die bestimmt welche Zustände die zugehörigen Zellen in der nächsten Generation annehmen. In der folgenden Generation werden die Blöcke dann neu, anders zusammengefasst.

Beim Typ "Block CA" in GeZA werden jeweils 2 Zellen zu einem Block zusammengefasst, in der folgenden Generation wird um die Zusammenfassung um eine Zelle geshiftet (siehe Grafik unten).



Der Modus "Margolus" in GeZA-2D ist auch ein Block-Zellularautomat!

Weitere Informationen finden Sie im Wikipedia.Eintrag: [block cell automaton \(englisch\)](#).

Optionen

totalistisch

Hier können Sie Ihre eigenen Zellularautomaten definieren!

Bei Regel ist eine Zeichenkette aus Ziffern und Sonderzeichen (+=-!?!><) anzugeben, diese ist Vorschrift für die Transformation einer Zelle zur nächsten Generation. Dazu steht Detailliertes in der Anleitung unter Grundlagen "totalistisch".

"max. Wert bei Zufall" legt die größte Zufallszahl bei der Generierung eines zufälligen Anfangszustands fest. Kommt in der Regel ein ? (zufälliger Wert in der nächsten Generation) vor, wird der maximale Wert ebenfalls durch diesen Parameter festgelegt.



"Totalistisch" bedeutet, dass die Werte der Nachbarn aufsummiert werden. Auf diese Summe wird dann die Transformationsregel angewandt.

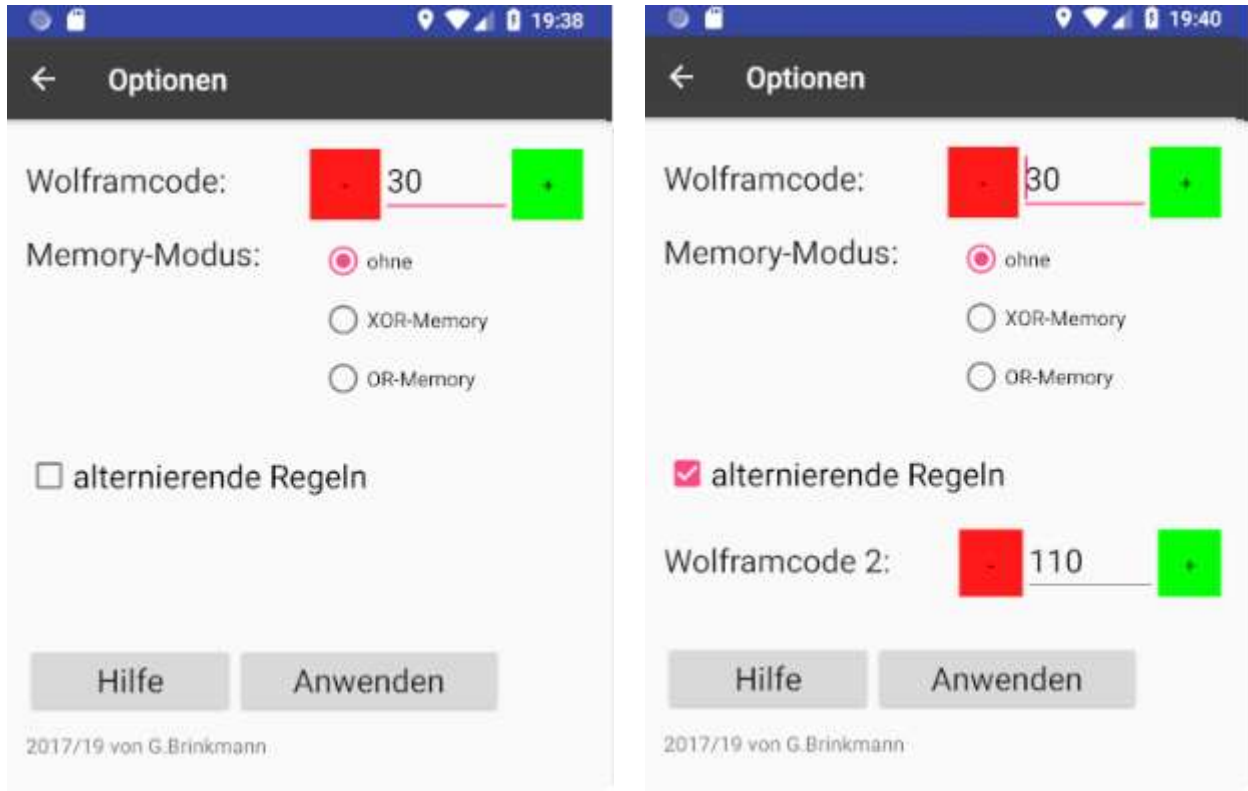
Die 5 Button hinter Nachbarn legen die Nachbarschaft fest: Grüne sind Nachbarn, graue nicht.

"0" ist die betrachtete Zelle selbst

Um den neuen Automaten zu testen auf "Anwenden" tippen.

Wolfram

Hier legt man die Nummer des Wolfram-Automaten fest. Er muss zwischen 0 und 255 liegen. Zusätzlich kann einer von drei Memory-Typen eingestellt werden. Details sind in der Anleitung unter "Grundlagen Wolfram" zu finden.



Aktiviert man die Check-Box "alternierende Regeln" so erscheint die Eingabe einer 2. Wolfram-Regel:

Der Automat benutzt dann abwechseln die beiden Regeln, der Memory-Modus wird immer benutzt.

Um den neuen Automaten zu testen auf "Anwenden" tippen.

NaSch-Modell

Die 2 Parameter Fahrzeugdichte und Trödelfaktor können hier geändert werden. Beides sind Prozentwerte und müssen zwischen 0 und 100 liegen. Details entnehmen Sie bitte der Anleitung (Kapitel "Grundlagen NaSch-Modell")



Um den neuen Automaten zu testen auf "Anwenden" tippen.

TwoSteps

"TwoSteps" funktioniert analog des Typs "totalistisch". Der Unterschied ist, dass für gerade und ungerade Generationen jeweils eine eigene Vorschrift definiert wird.

Hier können Sie Ihre eigenen Zellularautomaten definieren!

Bei Regel ist eine Zeichenkette aus Ziffern und Sonderzeichen (+=-!><) anzugeben, diese ist Vorschrift für die Transformation einer Zelle zur nächsten Generation. Dazu steht Detailliertes in der Anleitung unter Grundlagen "totalistisch" bzw. "TwoSteps".

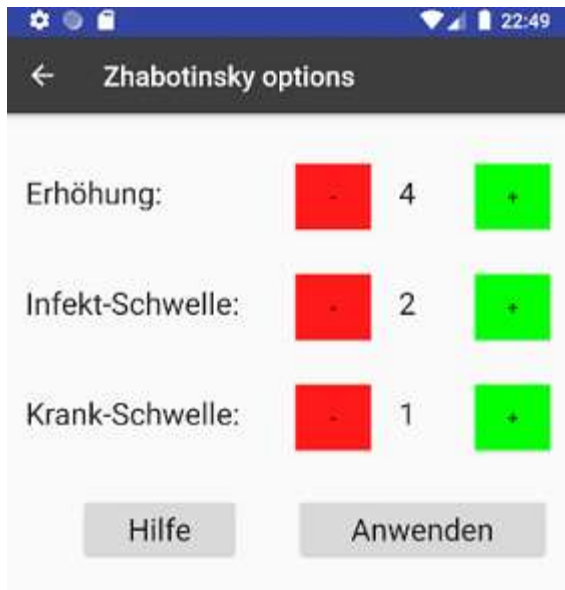


"Totalistisch" bedeutet, dass die Werte der Nachbarn aufsummiert werden. Auf diese Summe wird dann die Transformationsregel angewandt.

Die 5 Button hinter Nachbarn legen die Nachbarschaft fest: Grüne sind Nachbarn, graue nicht. "0" ist die betrachtete Zelle selbst.

Um den neuen Automaten zu testen auf "Anwenden" tippen.

Zhabotinsky



"Erhöhung" ist der konstante Wert (g) um den eine "infizierte" Zelle bei jedem Durchlauf erhöht wird.

"Infekt-Schwelle" ist die Mindestanzahl von infizierten Nachbarzellen um eine gesunde Zelle zu infizieren.

"Krank-Schwelle" ist die Mindestanzahl von kranken Nachbarzellen um eine gesunde Zelle zu infizieren.

"Erhöhung" kann zwischen 1 und 120 liegen.

"Infekt-Schwelle" und "Krank-Schwelle" können Werte zwischen 1 und 5 annehmen.

Optionen Block CA

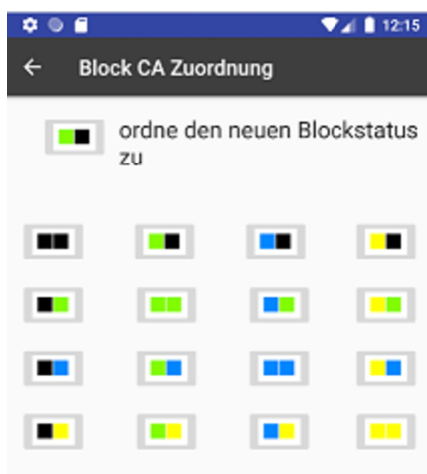
Hier kann die Regel für den Block-Automaten definiert werden!

Bei diesem Typ werden hier jeweils 2 Zellen, (Block) zusammen betrachtet. Die einzelne Zelle kann 4 Zustände annehmen. Die Regel sagt nun wie der Status dieser 2 Zellen dann in der neuen Generation ist.

Ein Block kann 16 verschiedene Zustände einnehmen. In diesem Dialog sehen Sie für diese 16 Blöcke die aktuelle Regel:



Nach anklicken einer dieser Zuordnungen öffnet sich ein weiteres Dialogfenster:



Hier wählt man den neuen Zustand für den gewählten Block und gelangt dann wieder zurück.

Credits



GeZA - App für Android



entwickelt 2016/19 von G. Brinkmann

Webside: www.zellauto.de

E-Mail: geza2d@zellauto.de

Danksagung:



Ohne die Unterstützung, das Verständnis, den Ansporn durch meine Frau Brigitte ist dieses Projekt nicht denkbar!

IHDL

WIKIPEDIA

Dank an das großartige Wikipedia-Projekt!

Literatur:

Gerd Brinkmann

Das Software-Experiment
PC Schneider International 2/1987 S.108ff.

Martin Gerhard/Heike Schuster

Das digitale Universum
Vieweg 1995

Stephen Wolfram

A New Kind of Science
Wolfram Media 2002

Jörg R. Weimar

Simulation with Cellular Automata
Logos Verlag 1997/2003
<http://www.jweimar.de/ZAscriptmml/gliederung.html>

Daniel Scholz

Pixelspiele
Springer Spektrum 2014

Links

www.cell-auto.com

www.mirekw.com

www.fourmilab.ch/cellab