

Anleitung: GeZA-2D für Android

Mit GeZA-2D können 2-dimensionale zelluläre Automaten untersucht und in attraktiven Grafiken dargestellt werden.

Inhaltsverzeichnis:

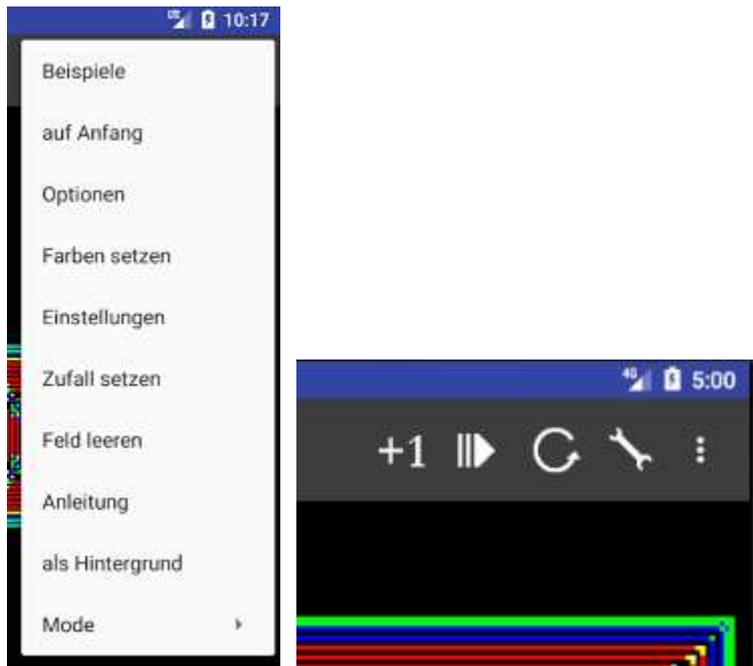
Bedienung	2
Allgemein.....	2
Regel-Editor.....	4
Regel-Konverter.....	6
Farben setzen.....	7
Einstellungen.....	8
Datei speichern.....	9
Datei öffnen.....	10
GIF erzeugen.....	11
Grundlagen.....	12
Standard.....	12
Hexagonal.....	14
WaTor (Water Torus).....	14
Brian's Brain.....	15
HPP Gas.....	15
Margolus (Block Zellularautomat).....	16
Zhabotinsky.....	17
Optionen.....	18
Standard.....	18
Hexagonal.....	19
WaTor.....	20
Brian's Brain.....	21
Margolus.....	22
Zhabotinsky.....	23
Credits.....	24
Links.....	25

Bedienung

Allgemein

Bei Start von GeZA2D baut sich der Standard-Automat auf oder der letzte Zustand wird angezeigt.

Sie können im Hauptfenster malen, d.h. Zellen einen neuen Wert geben. Weitere Programmoptionen finden sich im Drop-Down-Menü und in der Kopfzeile:

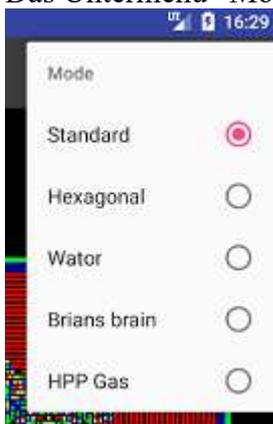


Je nach physikalischer Display-Größe werden nicht alle 4 Icons in der Titelzeile dargestellt, diese Funktionen können dann über das Drop-down-Menü angewählt werden.

Menü-Eintrag	Icon	Funktion
1 Schritt	+1	genau eine weitere Generation berechnen.
Start/Stop	Play/Pause	Starten bzw. stoppen der Animation.
Auf Anfang	Refresh	auf die Startbedingungen zurückstellen (stoppt laufende Animation).
Optionen	Wrench	es erscheint ein Dialogfenster, in dem kontextbezogenen Einstellungen vorgenommen werden können. (stoppt laufende Animation)
Beispiele	-	Öffnet die Liste der vordefinierten Automaten. Ändert ggf. den Modus. (stoppt laufende Animation)
Farben setzen	-	Farbvorgaben für die verschiedenen

		Zellenzustände verändern. (stoppt laufende Animation)
Einstellungen	-	öffnet ein Dialogfenster für globale Einstellungen. (stoppt laufende Animation)
Zufall setzen	-	füllt das "Spielfeld" zufällig. (stoppt laufende Animation)
Feld leeren	-	alle Zellen des "Spielfelds" werden auf den Wert 0 gesetzt. (stoppt laufende Animation)
Anleitung	-	Beschreibung ansehen. (stoppt laufende Animation)
als Hintergrund	-	das aktuelle Bild kann als Hintergrundbild (wallpaper) festgelegt werden. (stoppt laufende Animation)
Mode	-	Untermenü für die unterstützten Grundautomaten öffnen.

Das Untermenü "Mode":



Hier wird der Grundtyp des Zellularautomaten gesetzt.

Das Spielfeld ist in dieser App ein "geschlossenes", d.h. der obere Rand geht in den unteren und der linke in den rechten Rand über!

Topologisch spricht man von einem Toroid. Anschaulich vorstellen kann man sich das z.B. als Fahrradschlauch.

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
0 .. 9	Die Zelle erhält den angegebenen Wert 0..9
=	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.

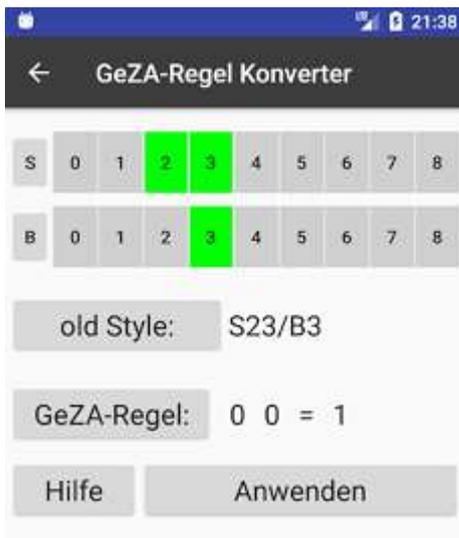
Hinweis: Bei sehr langen Regel-Strings gibt es Anzeigeprobleme. Es ist ein bekanntes Problem!

Regel-Konverter

Häufig findet man in der einschlägigen Literatur oder auch Software (meistens zum Thema "Game of Life") Regelangaben in der Form 23/3 oder S23/B3 oder ähnlich. S bedeutet stabil



und B Geburt (Birth). Hier wird für die Moore-Nachbarschaft gesagt, das die Zelle bei 2 lebenden (Wert=1) Nachbarn stabil und bei 3 lebenden Nachbarn die betrachtete Zelle weiterhin lebt bzw. "geboren" wird. Dies ist im Übrigen die Regel vom Game of Life. Da GeZA2D eine eigene neue Art von Regel-Konvention benutzt, gibt es dieses kleine Tool um Regeln dieser alten Schreibweise in die GeZA-Form zu konvertieren.



Sie können die Schalter (0..8) neben dem S (stabil) und B (Birth) aktivieren oder deaktivieren. Bei old Style steht die jeweilige Regel in der konventionellen Schreibweise und bei GeZA-Regel in der von dieser App benutzten.

Durch Klicken auf den Button "Anwenden" gelangt man zurück in den Optionen-Dialog mit der hier definierten Regel.

Achtung! Ist die bestehende Regel beim Aufruf des Konverters nicht mit in der "Old-Style"-Form abbildbar, so wird die "Game of Life"-Regel (00=1) vorgegeben.

Farben setzen

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



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.

Einstellungen

Hier können Einstellungen vorgenommen werden, die unabhängig vom gerade ausgewählten Modus sind.



Parameter	Funktion
Zellengröße	gibt die Darstellungsgröße einer Zelle an. Zulässige Werte liegen zwischen 2 und 7. Bei 2 hat eine Zelle 2x2 Pixel, bei 7 25x25 Pixel. Je kleiner die Zelle umso größer die Zahl der zu berechnenden Zellen, was das Programm sehr langsam machen kann.
Speed	Geschwindigkeit der Generationenberechnung, Zulässige Werte sind 1 (sehr langsam) bis 8 (am schnellsten) .
max. Zufallswert	Beim zufälligen Setzen von Zellen wird maximal dieser Wert gesetzt, also Werte von 1 bis max. Zufallswert.
Edit-Farbe	Beim Malen auf den Bildschirm erhalten die neuen Pixel diesen Wert, bzw. die Farbe, die diesem Wert entspricht. Zulässig sind 1 bis 9.

Datei speichern

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



The screenshot shows a file saving dialog with the following elements:

- Verzeichnis: /storage/emulated/0/GeZA2D
- Datei (png): geza_2021-04-05_141047.png
- Datei (GeZA-2D): geza_2021-04-05_141047.gz2
- Buttons: 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.

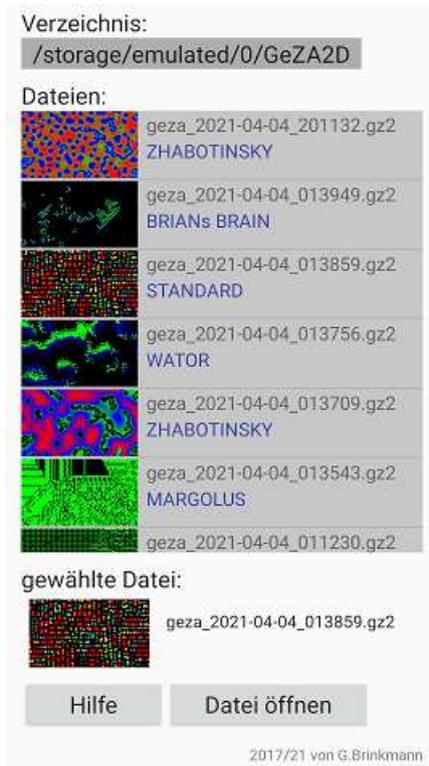
Man hat die Möglichkeit 2 Typen von Dateien zu speichern:

- 1) Eine Grafik des aktuellen Bildschirms im PNG-Format
- 2) den aktuellen Zellularautomaten mit allen Parametern und Einstellungen im GZ2-Format, Diese Datei kann nur mit der GeZA-App wieder eingelesen werden, aber problemlos auf anderen Android-Geräten mit der GeZA2D-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.

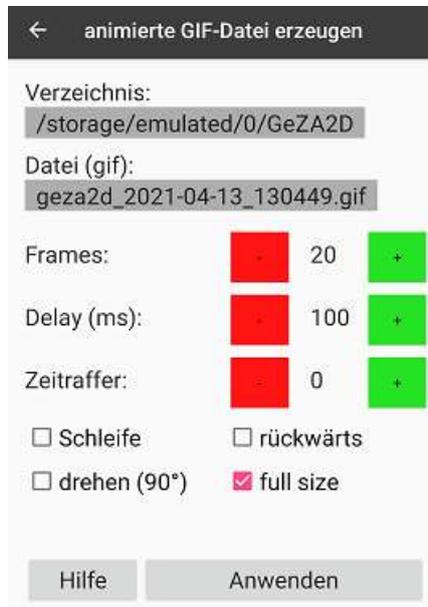


Die GeZA2D-Dateien werden in einem fest vorgegeben Verzeichnis gesucht. Dieses Verzeichnis ("GeZA2D" auf dem internen Speicher) kann noch nicht frei gewählt werden. Der vollständige Name wird in der ersten Zeile angezeigt.

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.

GIF erzeugen

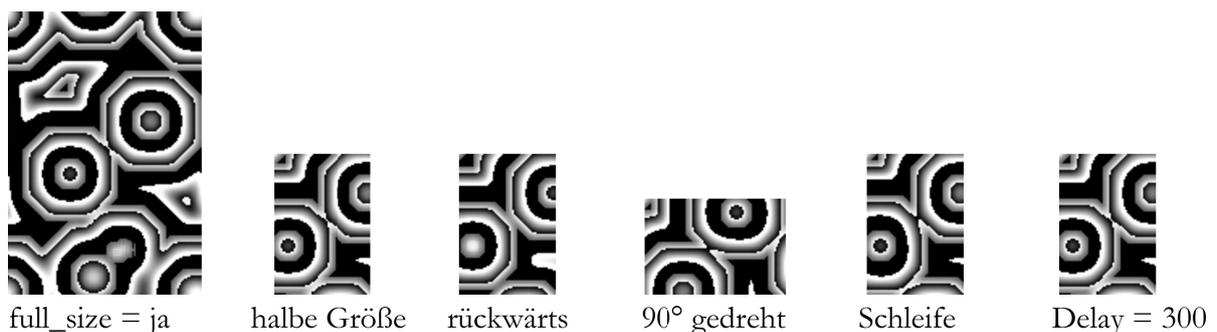
Hier kann mit dem aktuellen Zellularautomaten eine animierte GIF-Datei generiert werden!.



Die Bedeutung der Parameter und Optionen:

Parameter/ Option	Bedeutung	mögliche Werte
Frames	Anzahl der Bilder	2 - 200
Delay	Pause zwischen 2 Bildern in ms.	10 - 2000
Zeitraffer	nach jedem Bild werden jeweils diese Anzahl Bilder übersprungen	0 -20
Schleife	nach dem normalen Ablauf wird der gleiche Ablauf in umgekehrter Reihenfolge angehängt	ja/nein
rückwärts	die Animation läuft rückwärts ab	ja/nein
drehen (90°)	die Animation wird um 90° nach links gedreht.	ja/nein
full size	voll Bildschirmgröße oder halbe Höhe und Breite	ja/nein

Hier die Auswirkungen der obigen Parameter am Beispiel:



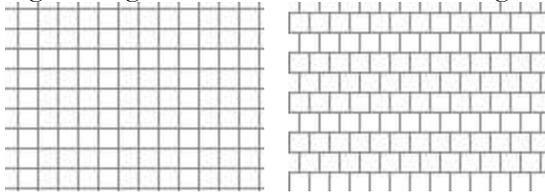
Grundlagen

Standard

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-2d behandelt ausschließlich 2-dimensionale Zellularautomaten, d.h. die Zellen sind regelmäßig auf einer "Gitter"-Fläche angeordnet.



Die linke Abbildung zeigt eine schachbrettartige Anordnung und ist hier bei den Modus Standard, Wator, Brians Brain und HPP-Gas die Grundlage.

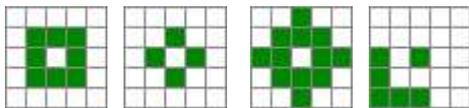
Die rechte Abbildung ist ein verzerrtes Gitter und hat logisch ein wabenartiges/sechseckiges Verhalten und ist hier die Grundlage für den Hexagonal Modus.

Man kann sich noch viele weitere Parkettierungen vorstellen. Im Folgenden wird der Standard-Modus behandelt.

Die einzelnen Zellen auf dem Spielfeld können diskrete Zustände annehmen, in dieser App die Werte zwischen 0 und 9.

Diese Werte werden bei der Darstellung farbige dargestellt. (0 = schwarz, 1 = grün, 2 = blau, usw.) Die Farbgebung kann angepasst werden.

Desweiteren muss eine Nachbarschaft definiert sein:



Auf dem linken Bild ist zu erkennen, dass die Zelle 8 Nachbarn (grün) besitzt. Dies ist unter dem Namen Moore-Nachbarschaft bekannt und wird beim oben schon erwähnten Game of Life verwendet.

Das zweite ist die sogenannte von Neumann-Nachbarschaft. Die Nachbarschaft kann auch aus einem weiteren Umfeld der Zelle kommen und die Zelle selbst kann ebenso zur Nachbarschaft gehören.

In dieser App können Sie die Nachbarschaft frei definieren.

Nun fehlt noch die Transformationsvorschrift (Regel), welche bestimmt wie sich die Werte der Zellen beim Übergang in die nächste Generation ändern.

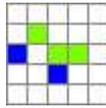
Diese App verwendet im Standard- und Hexagonal-Modus sogenannte "totalistische" Regeln.

Dabei werden die Werte aller Nachbarzellen aufsummiert. Anhand dieser Summe wird über eine Regeltabelle der neue Wert der Zelle festgelegt.

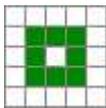
Hier am Beispiel:

Summe	0	1	2	3	4	5	6	7	8	9	...
Nachbarwerte											
neuer Wert	0	2	1	4	3	+	=	0	0	0	...

Diese Regel in Kurzform geschrieben : 02143+=

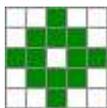


Anfangszustand: (0=weiß, 1=grün, 2= blau)



Nachbarschaft:

Die Nachbarschaft der mittleren grünen Zelle besteht aus 2 grünen, einer blauen und 5 weißen Zellen. Die Summe der Nachbarzellen ($5 \cdot 0 + 2 \cdot 1 + 1 \cdot 2$) ist somit 4. In der Regeltabelle steht für die Summe 4 der Wert 3. Diesen Wert 3 hat also die betrachtete Zelle in der Folgegeneration.



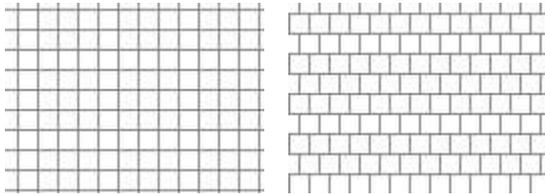
Nimmt man als Nachbarschaft, so hat die gleiche Zelle 8 weiße, 2 grüne und 2 blaue Nachbarzellen. Die Summe ist demnach 6. In der Regeltabelle finden wir für Summe 6 das Gleichheitszeichen=. Dies bedeutet: die betrachtete Zelle behält ihren Wert, also ist sie in der Folgegeneration ebenfalls grün.

Der Tabelle unten kann man entnehmen welche Funktionszeichen es neben den Ziffern gibt:

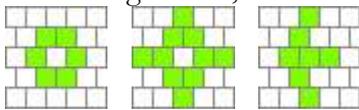
Zeichen	Bedeutung
=	Die Zelle behält ihren Wert.
+	Der Wert der Zelle wird um 1 erhöht, aus 9 wird Null.
-	Der Wert der Zelle wird um 1 erniedrigt, Null bleibt Null.
!	Aus Null wird 1, ansonsten erhält die Zelle den Wert Null.
?	Die Zelle erhält einen zufälligen Wert.
>	Die Zelle erhält den größten Nachbarzellenwert.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

Hexagonal

Der einzige Unterschied zum Standard-Modus ist, dass das Grundgerüst nicht schachbrettartig (linkes Bildchen, sondern eine versetzte Struktur hat, was logisch einer wabenförmigen Struktur entspricht.



Daraus ergibt sich, aber auch, dass die Nachbarschaft entsprechend anders aussieht.



Alles andere ist analog zum [Standard-Modus](#).

WaTor (Water Torus)

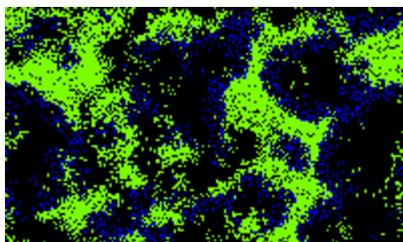
Der Begriff Wator leitet sich von Water Torus ab, also von Wasser und Toroid (eine in sich geschlossene 2-dimensionale Fläche, ein Fahrrad-Schlauch ist z.B. eine solche).

Diesen Wasser-Toroid kann man sich als Meer vorstellen, in dem es Fische und Haie gibt. Die Fische vermehren sich nach bestimmten Regeln, die Haie fressen die Fische um zu leben und sich vermehren zu können.

Gibt es wenig Haie, so nimmt die Population der Fische stark zu. Dies führt dazu, dass die Haie viel Nahrung haben und sich ihrerseits stark vermehren, aber dabei ihre Lebensgrundlage, den Fischbestand, vernichten.

Der geringe Fischbestand führt jetzt dazu, dass die Haie verhungern. Und so weiter.

An Wator kann man sehr schön die Entwicklung einer [Jäger-Beute Beziehung](#) sehen. Es gehorcht den [Loitka-Volterra-Regeln](#).



Für weitergehende Informationen siehe den zugehörigen Wikipedia-Eintrag:
<https://de.wikipedia.org/wiki/Wator>

Brian's Brain

Dies ist ein spezieller von Brian Silverman erdachter zellulärer Automat.
 Mehr Informationen hierzu finden Sie im zugehörigen Wikipedia-Eintrag (englisch):
https://en.wikipedia.org/wiki/Brian%27s_Brain

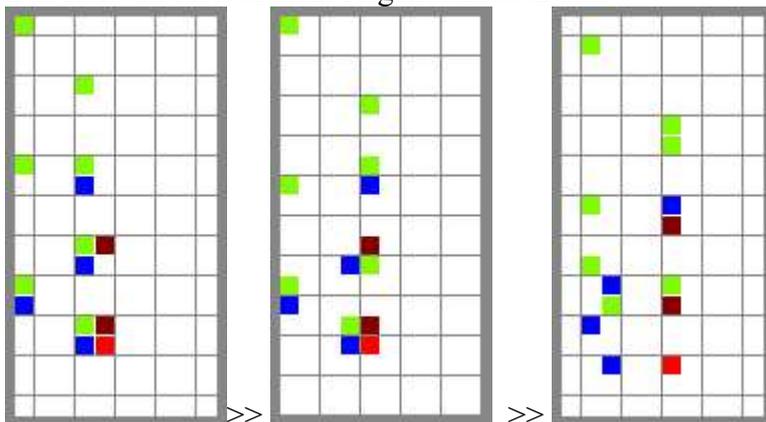
HPP Gas

Dieser Automat benutzt nicht die bekannte Nachbarschaft, sondern fasst 4 Zellen zusammen (in den Randbereichen auch 1 und 2 Zellen).
 Der folgenden Tabelle sind die Übergangsvorschriften zu entnehmen. Durch Rotation erhält man alle Möglichkeiten.

vorher	nacher

In der darauf folgenden Generation wird die Zusammenfassung der 4 Zellen um 1 Zeile und 1 Spalte verschoben.

Das sieht im Ablauf dann folgendermaßen aus:



Dieses Verfahren ist als Margolus-Nachbarschaft bekannt.

Weitere Information entnehmen Sie bitte dem Wikipedi-Eintrag „lattice gas automaton“ (englisch): https://en.wikipedia.org/wiki/Lattice_gas_automaton

Margolus (Block Zellularautomat)

Mehr Informationen hierzu finden Sie im zugehörigen Wikipedia-Eintrag (englisch) und bei [Mirek Wojtowicz](#) (englisch) bzw. Tim Tyler:

https://en.wikipedia.org/wiki/Block_cellular_automaton

http://www.mirekw.com/ca/rullex_marg.html

<http://cell-auto.com/neighbourhood/margolus/index.html>

In diesem Modus kann eine Zelle nur die Zustände 0 und 1 annehmen. Um schwierige Wandbehandlungen zu vermeiden ist die "Spielfläche" ein geschlossener Toroid und besteht aus einer geraden Zahl von Zeilen und Spalten.

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.

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 $\lfloor [K/k_1] + [I/k_2] \rfloor$ zugewiesen.

wobei K die Anzahl der kranken Nachbarn und I die Anzahl der infizierten Nachbarn ist. k_1 ist die Krank-Schwelle (1-5), k_2 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. $[S/A]$ 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).

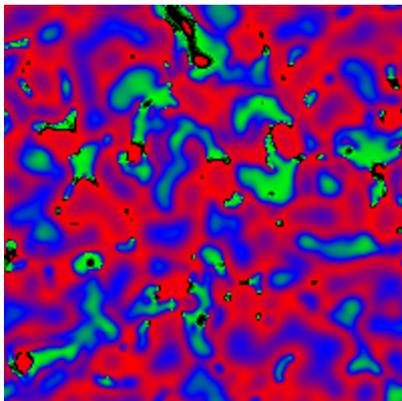
Bei diesem Automaten wird die Moore-Nachbarschaft ($r=1$)  verwendet. Die Simulation läuft auf einer geschlossenen Fläche (Toroid) ab.

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 k_1 , k_2 , und g, sowie den Anfangsbedingungen, entstehen.

Unten sind 2 typische Beispiele:



$k_1=3, k_2=2, g=5$



$k_1=3, k_2=2, g=64$

Optionen

Standard

Hier können die beiden bestimmenden Eigenschaften des Standard-Zellularautomaten eingestellt werden. Zum Einen die Transformationsregel und zum Anderen die Nachbarschaft.



Durch Klicken auf den Regel-String wird der Regel-Editor geöffnet.

Die Button 0 bis 24 können durch Touch zwischen grau und grün umgeschaltet werden. Die grünen geben die aktuelle Nachbarschaft an.

Mit dem Button "Regel-Konverter" können konventionelle Regeln in die, von GeZA2d verwendeten, konvertiert werden.

Hexagonal

Hier können die beiden bestimmenden Eigenschaften eines "hexagonalen Zellularautomaten" eingestellt werden. Zum Einen die Transformationsregel und zum Anderen die Nachbarschaft.



Durch Klicken auf den Regel-String wird der Regel-Editor [geöffnet](#).

Die Button 0 bis 18 können durch Touch zwischen grau und grün umgeschaltet werden. Die grünen geben die aktuelle Nachbarschaft an.

WaTor

In diesem Modus können 4 Parameter angepasst werden.



Parameter-Name	Bedeutung
Fisch Repro-Zyklus	Anzahl der Generationen nach dem sich ein Fisch vermehrt (reproduziert).
Fisch Lebenszeit	der Fisch stirbt, wenn er soviel Generationen gelebt hat.
Hai Repro-Zyklus	Anzahl der Generationen nach dem sich ein Hai vermehrt (reproduziert).
Hai Hungerzyklus	hungert ein Hai mindestens soviel Generationen, stirbt er.

Hinweis: ein Fisch-Repro-Zyklus größer als die Fisch-Lebenszeit macht kein Sinn, da der Fisch stirbt bevor er sich vermehren kann.

Brian's Brain

Bei Brian's Brain gibt es nur einen Parameter an dem gedreht werden kann:



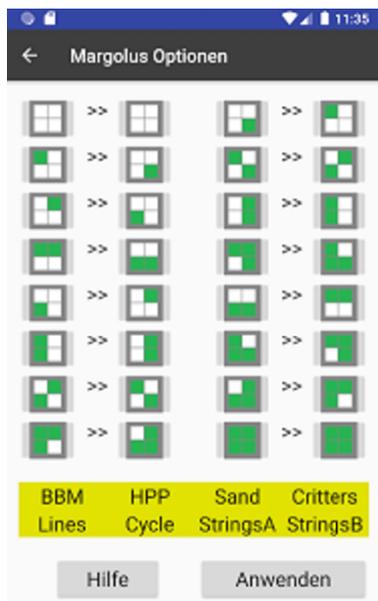
"Brain Max" legt den Wert fest, den eine Zelle erhält, wenn Sie genau 2 Nachbarn mit dem Wert "Brian Max" hat. Standard ist 2.

Margolus

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

Bei diesem Typ werden ja immer jeweils 4 Zellen (Block) zusammen betrachtet, die Regel sagt wie der Status dieser 4 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. Es gibt 8 Schaltflächen mit denen jeweils eine vordefinierte Regel gesetzt werden kann.

Zhabotinsky

← Zhabotinsky Optionen

Erhöhung: 5

Infekt-Schwelle: 2

Krank-Schwelle: 1

Nachbarn:

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24

Hilfe Anwenden

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

"Infekt-Schwelle" ist die Mindestanzahl von infiziert 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.

Ebenso kann hier die Nachbarschaft frei definiert werden. Grün bedeutet, dass die entsprechende Zelle berücksichtigt wird.,

Weiteren Informationen zum Algorithmus entnehmen Sie bitte den „Grundlagen Zhabotinsky“

Credits

GeZA-2D - App für Android



entwickelt 2016/20 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