Complex Systems Engineering

Zelluläre Automaten

Prof. Dr. Christin Seifert

7. November 2017

University of Passau, WS 2017/2018

Überblick

1. Einführung
2. Modell
3. Eindimensionale Automaten
4. Zweidimensionale Automaten
5. Anwendungsbeispiele
6. Zusammenfassung

Einführung

Zelluläre Automaten

- Mathematische Modelle mit diskretem Raum und diskreter Zeit
- Zur Modellierung räumlich diskreter dynamischer Systeme
- Zellen haben feste Nachbarschaft
- Es gibt Regeln für den Zustand von Zellen
- Synonyme: Zellularautomaten, Polyautomaten, cellular automata, CA

Geschichte

- In den 1940er Jahren erstmals von Stanislaw Ulam vorgestellt
- Studie zum Wachstum von Kristallstrukturen
- Raum war ein einfaches Gittermodell

Abbildung 1: Stanislaw Ulam und John von Neumann (Public Domain via Wikimedia Commons)

Geschichte

- John von Neumann generalisiert Ulams Idee zu einem allgemeinen Berechnungsmodell.
- Ca. 1950 Modell zur Berechnung der Bewegung von Flüssigkeiten von Ulam und von Neumann. Flüssigkeit wird aus einzelne Elemente bestehende betrachtet; Bewegung der Elemente ergibt sich aus der Bewegung ihrer Nachbarzellen.
- Ende der 1960er Jahre ”Game of Life”von John H. Conway. Zweidimensionales Gitter, in dem Zellen leben und sterben können. Zustand wird durch einfache Regeln (Zustand der Nachbarn) bestimmt. Siehe Folie 36.
- 1969 vertritt Konrad Zuse die Annahme, dass Naturgesetze diskreten Regeln folgen und somit der Ablauf des Universums einem Zellularautomaten entspricht.
- 2002 bewies Stephen Wolfram, dass eine Turing-Maschine auf Basis des elementaren zellulären Automaten mit der Nummer 110 gebaut werden kann. Siehe Folie 21.

Definition

Definition Zellulärer Automat

Unter einem zellulären Automaten (CA) versteht man eine (meist) zweidimensionale, gitterförmige Anordnung quadratischer Zellen nebst zugehöriger Regeln, die beschreiben, in welcher Weise die Zustände der Nachbarzellen den Zustand einer Zelle beeinflussen [2].

- Zur Modellierung von räumlich und zeitlich diskreten, dynamischen Systemen
- Simulation von dynamischen Strukturen, um Rückschlüsse auf Phänomene in der realen Welt zu ziehen
- Alle Zellen sind gleich und folgen gleichen Entwicklungsregeln
- Entwicklung einzelner Zellen unwichtig
- Gesamtheit des Entwicklungsprozesses und Strukturen, die über die Zeit entstehen, sind von Interesse

Modell

Definition Zellulärer Automat

Ein zelluläre Automat (CA) ist definiert durch das Tupel (L, N, S, f), d.h.

$$CA = (L, N, S, f)$$

Die Komponenten haben folgende Bedeutung:

- Zellraum L: Beschreibung des Gitters
- Nachbarschaft N: Beschreibung der Umgebung einer Zelle
- Zustandsmenge S: Menge der Elementarzustände für eine Zelle
- Lokale Übergangsfunktion f: Regeln für Zustandsübergänge einer Zelle

Die Symbole stammen von den englischen Bezeichnungen ab:
L - Lattice, N - Neighborhood, S - States

Zellraum L definiert durch

- Größe
  - theoretisch unendlich, für praktische Simulation jedoch endlich
  - in endlichen Gittern Festlegung von Randbedingungen (siehe Folie 12)
- Dimensionalität
  - meist zweidimensional, jedoch auch eindimensionale Automaten (Linien) und dreidimensionale Automaten (Räume) möglich
- Geometrie
  - meist regelmäßiges Gitter der Größe $n \times m$ (n, m Variablen)

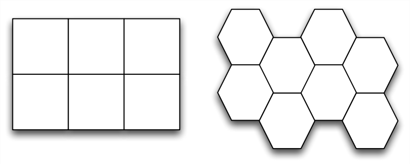


Abbildung 2: Beispiele für regelmäßige Gitter

Nachbarschaft N definiert, welche Zellen die aktuelle Zelle beeinflussen können

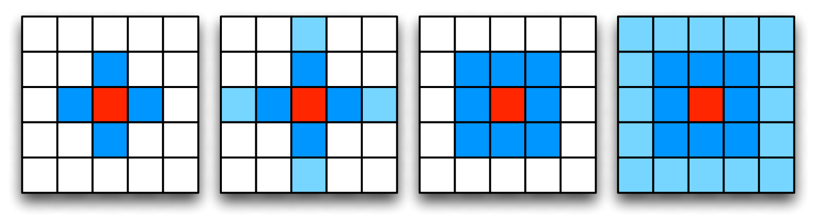
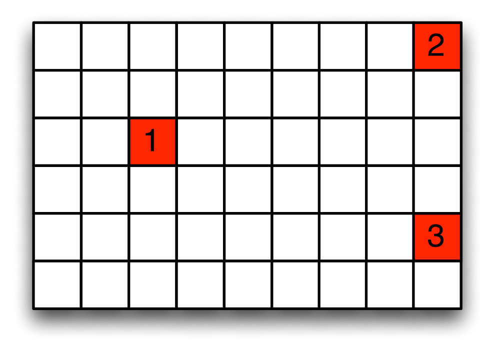


Abbildung 3: Beispiele für Nachbarschaften. Rot ist die aktuelle Zelle, blau sind deren Nachbarn. Von Neumann, erweiterte von-Neumann, Moore und erweiterte Moore Nachbarschaft (v.l.nr.).

Aufgabe

Bestimmen Sie die von-Neumann und Moore-Nachbarschaft im u.g. endlichen Gitter für die Zellen 1, 2, und 3.
Wieviel Nachbarn haben die Zellen jeweils?



In endlichen Gittern können Zellen unterschiedlich viele Nachbarn haben.

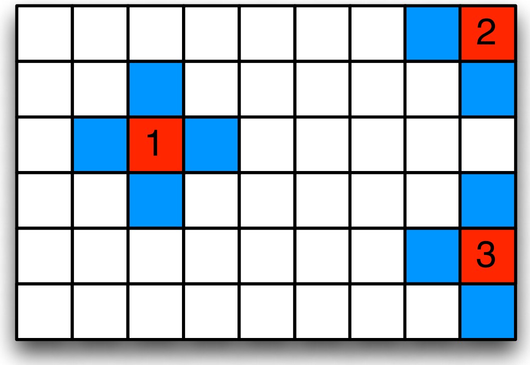


Abbildung 4: Zelle 1 hat vier, Zelle 2 hat zwei, Zelle 3 hat drei Nachbarn.

Nachbarschaft N

Randbedingungen zur Auflösung des Nachbarschaftsproblems

- Reflexive Randform: als Zustand der nicht-existierenden Nachbarn wird der Zustand der Zelle selbst angenommen

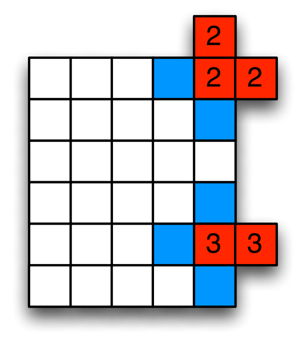


Abbildung 5: Reflexive Randform

NACHBARSCHAFT N

Randbedingungen zur Auflösung des Nachbarschaftsproblems

- Periodische Randform: flaches Gitter ringförmig schließen

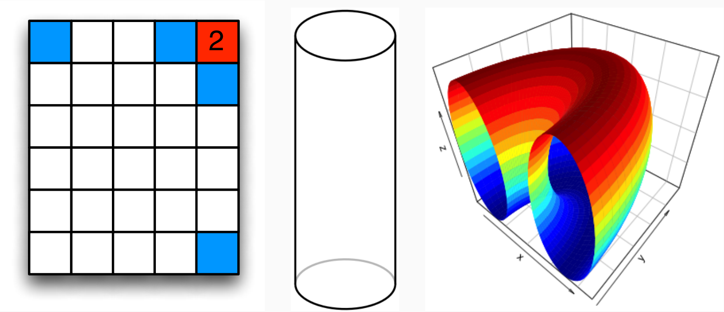


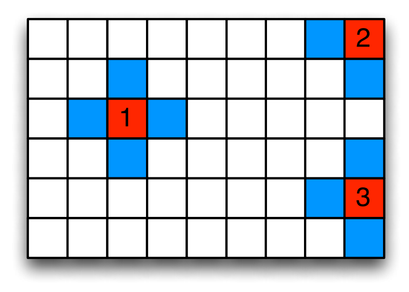
Abbildung 6: Nachbarschaft für Zelle 2 bei Ringschluss horizontal und vertikal. Modell Zylinder (Ringschluss horizontal) und Torus (Ringschluss horizontal und vertikal), v.l.n.r.

ZUSTANDSMENGE S

- Zustandsmenge ist diskret, wenige Zustände, oft $|S| = 2$
- Anzahl Zustände eines Automaten mit N Zellen und k Zuständen ist k^N
- Automaten haben endlich viele Zustände (da endliches Gitter, damit endlich viele Zellen und endlich viele Zustände)
- Zustände lassen sich damit durch natürliche Zahlen angeben

Aufgabe

1. Wieviel Zustandskombinationen $(z_1, N(z_1))$ gibt es im u.g. Gitter für Zelle 1, wenn jede Zelle genau 2 Zustände annehmen kann?
2. Wieviel verschiedene Funktionen $f(z_1, N(z_1))$ lassen sich für die Zustandsänderung von Zelle z_1 definieren?



1. $2^5 = 32$
2. $2^{2^5} = 4.294.967.296$

Lokale Übergangsfunktion f

- Sei i eine beliebige Zelle (d.h. ein Gitterpunkt) und $N(i)$ deren Nachbarschaft
- Zustand der Zelle z_i hängt nur von ihrem eigenen Zustand und dem ihrer Nachbarn ab, d.h. $f = f(z_i, N(z_i))$
- Für einen Automaten mit $|S| = k$ und N Nachbarn pro Zelle gibt es

  $$k^{k^{N+1}}$$

  mögliche Regeln für Zustandsübergänge (d.h. verschiedene f)
- Intuitiv sind das $|output|^{|input|}$ viele Möglichkeiten, wobei $|output|$ die Anzahl möglicher Outputs angibt und $|input|$ die Anzahl möglicher Inputs.

Vorteile des Modells

CA in a Nutshell

$CA = (L, N, S, f)$

- Einfache Regeln erlauben Simulation von sehr komplexem Verhalten.
- Zustandsübergänge sind exakt (da diskret), d.h. es gibt kein Problem mit Rundungsfehlern o.ä.
- Einfache Implementierung und Darstellung (im 1D oder 2D Fall).
- Gute Repräsentation von empirischem Wissen, das sich of nicht in mathematischen Gleichungen einfach ausdrücken lässt.

Eindimensionale Automaten

Modulo-Automat

Beschreibung

- 1-dimensionales Gitter, 2 mögliche Zellzustände, $Z = \{0, 1\}$
- Zustandsübergangsfunktion der Zelle z_i von Zeitpunkt t auf Zeitpunkt $t + 1$ definiert als

$$z_i(t+1) = \begin{cases} 0 & \text{für } z_{i-1}(t) + z_{i+1}(t) \text{ gerade} \\ 1 & \text{für } z_{i-1}(t) + z_{i+1}(t) \text{ ungerade} \end{cases}$$

oder (gleichbedeutend)

$$z_i(t+1) = z_{i-1}(t) + z_{i+1}(t) \mod 2$$

Beschreibung

- 1-dimensionales Gitter, 2 mögliche Zellzustände, $Z = \{0, 1\}$
- Zustandsübergangsfunktion der Zelle z_i von Zeitpunkt t auf Zeitpunkt $t+1$ definiert als

$$z_i(t+1) = z_{i-1}(t) + z_{i+1}(t) \mod 2$$

Was ist der Zustand des Automaten auf einem Gitter mit 21 Zellen, bei dem $z_{11}(0) = 1$ und $z_i(0) = 0, \forall i \neq 11$ zum Zeitpunkt $t = 1, 2, \ldots$? Zeichnen Sie die Zustände.

MODULO-AUTOMAT

Zustand des Automaten auf einem Gitter mit 21 Zellen, bei dem $z_{11}(0) = 1$ und $z_i(0) = 0$, $\forall i \neq 11$ zum Zeitpunkt $t = 1, 2, \ldots$?

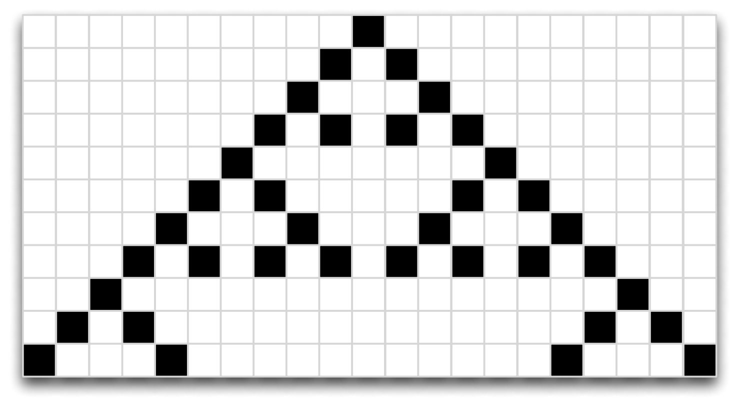


Abbildung 7: Zustände des Modulo-Automaten. Horizontale Dimension: Raum, Vertikale Dimension: Zeit ($t = 0$ oben)

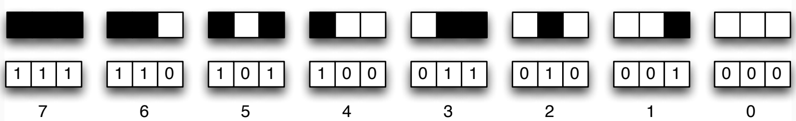
Beschreibung

- 1-dimensionales Gitter, periodische Randform
- 2 mögliche Zellzustände, $Z = \{0, 1\}$
- Zustandsübergangsfunktion der Zelle z_i von Zeitpunkt t auf Zeitpunkt $t + 1$ definiert als

$$z_i(t+1) = f(z_{i-1}(t), z_i(t), z_{i-1}(t))$$

Im Allgemeinen müssen die Regeln von f für jede Zellkombination einzeln kodiert werden (im Gegensatz zum Modulo-Automaten, der nur eine Regel umfasst).

- Da der Zustand einer Zelle nur von ihrem Zustand und der linken und rechten Nachbarzelle abhängt, gibt es 8 verschiedene Eingangsmuster



- $42_{10} = 00101010_2$

Wolframs Universum - Übergangsfunktion

- Es werden jeweils die Nachbarn und die Zelle z_i zum Zeitpunkt t und die Zelle z_i zum Zeitpunkt $t+1$ dargestellt
- Reihenfolge der Zustände der Zelle z_i wird als Binärzahl interpretiert und dann in eine Dezimalzahl umgewandelt, die den Automaten eindeutig spezifiziert
- Es gibt 256 verschiedene Automaten (entspricht der Anzahl der Zahlen, die sich mit 8 Bit darstellen lassen)

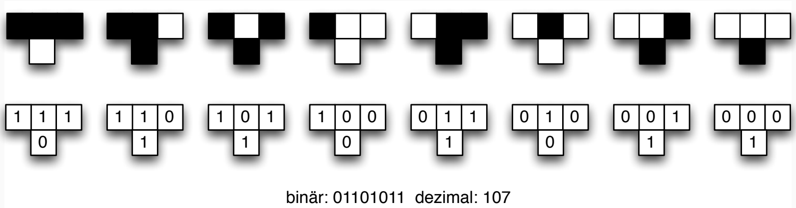


Abbildung 8: Regel 107

Aufgabe

- Zeichnen Sie die Regel für den Wolfram-Automaten Nummer 42.

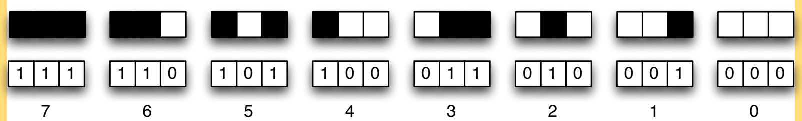


Abbildung 9: Eingangsmuster farbcodiert, als Binär- und als Dezimalzahl

Wolframs Universum – Regel 107

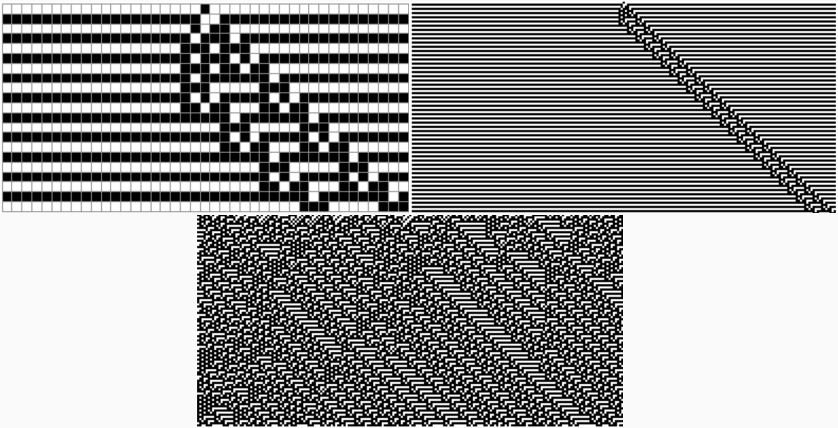


Abbildung 10: Ergebnisse für Regel 107. Oben: mit fixer Initialisierung zum Zeitpunkt 0 nach 20 bzw. 100 Iterationen. Unten: mit zufälliger Initialisierung
Erstellt mit Wolfram Alpha `http://www.wolframalpha.com/input/?i=rule+107`

Wolframs Universum – Regel 30

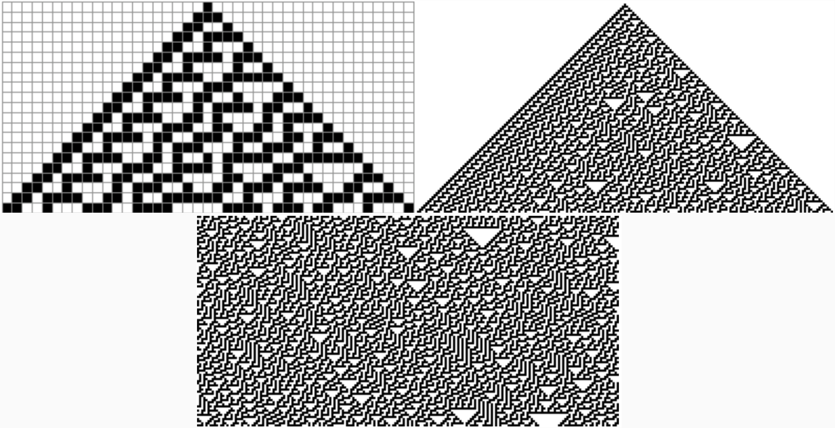


Abbildung 11: Ergebnisse für Regel 30. Oben: mit fixer Initialisierung zum Zeitpunkt 0 nach 20 bzw. 100 Iterationen. Unten: mit zufälliger Initialisierung
Erstellt mit Wolfram Alpha `http://www.wolframalpha.com/input/?i=rule+30`

Wolframs Universum – Regel 200

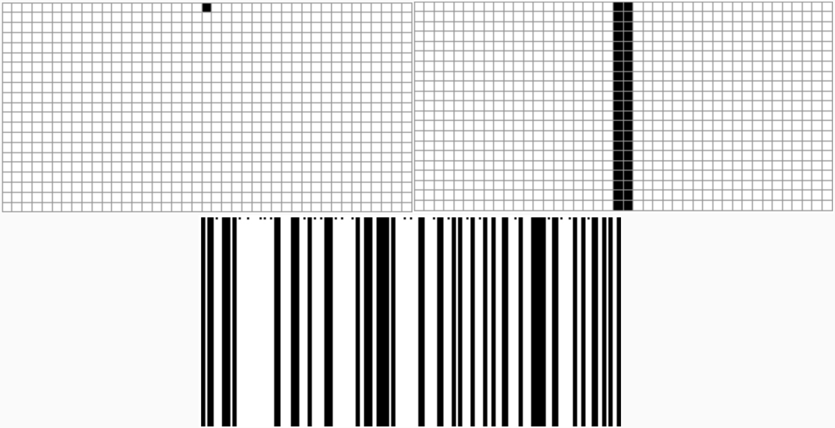


Abbildung 12: Ergebnisse für Regel 200. Oben: mit 2 verschiedenen, fixen Initialisierung zum Zeitpunkt 0 nach 20 Iterationen. Unten: mit zufälliger Initialisierung Erstellt mit Wolfram Alpha
`http://www.wolframalpha.com/input/?i=rule+200`

Wolframs Universum – Klassen

- **Klasse 1** (Fixpunkt-Zustand): Entwicklung hin zu einem räumlich homogenen Zustand ohne Muster, der unveränderlich ist. Beispiel: Regel 32
- **Klasse 2**: Entwicklung hin zu sich ständig wiederholenden oder periodischen Strukturen, Zustände werden irgendwann immer wiederholt. Beispiel: Regel 56
- **Klasse 3:** Entwicklung von chaotischem, aperiodischem Verhalten, ständige Veränderung.Beispiel: Regel 22
- **Klasse 4:** Keine periodische Wiederholung des Gesamtmusters, aber komplexe, lokale Strukturen, die sich in verschiedenen Größen wiederholen. Beispiel: Regel 110

Klasse 1 entspricht vollständiger Ordnung, während Klasse 3 vollständigem Chaos (unvorhersehbare Dynamik) entspricht. Klasse 4 ist zwischen Ordnung und Chaos. Hier kann Information in Raum und Zeit weitergegeben werden und es sind beliebige Rechnungen möglich (für Regel 110 wurde das von Matthew Cook gezeigt).

Wolframs Universum – Klassen

Abbildung 13: Klasse 1, Regel 32

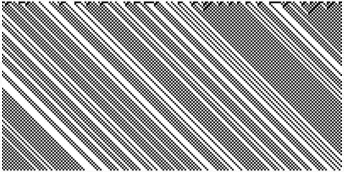


Abbildung 15: Klasse 2, Regel 56

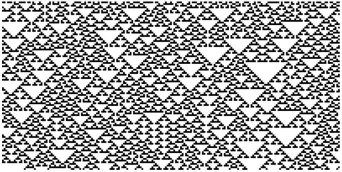


Abbildung 14: Klasse 3, Regel 22

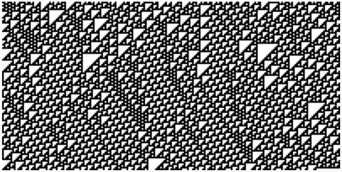


Abbildung 16: Klasse 4, Regel 110

Gibt es ein Maß, das für einen Automaten die Klasse vorhersagen kann? D.h. ein Maß, das vorhersagen kann, wie komplex das Verhalten des Automaten über die Zeit ist?

- Betrachtet wird die Übergangsfunktion
- Sei k die Anzahl der Zustände und p die Anzahl der Nachbarn einer Zelle (inkl. der Zelle selbst)
- Dann ist k^p die Anzahl aller möglichen Nachbarzustände
- Sei n_q die Anzahl der inaktiven Ergebniszustände

Definition Langton Parameter

Der Langton-Parameter λ ist definiert als

$$\lambda = \frac{k^p - n_q}{k^p}.$$

Der Langton-Parameter gibt somit den Anteil der überlebenden Zellen in der Übergangsfunktion an.

Wolframs Universum – Langton Parameter

Beispiel Regel 107

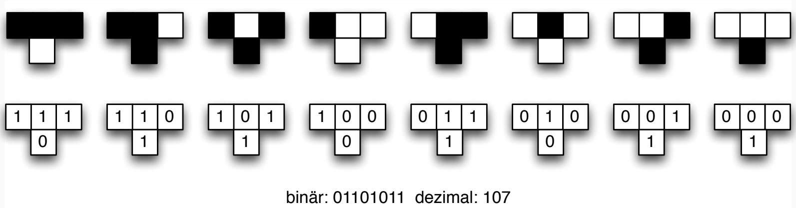


Abbildung 17: Regel 107

$k^p = 2^3 = 8$, $n_q = 3 \rightarrow \lambda = \frac{8-3}{8} = 0.625$

Aufgabe

Berechnen Sie den Langton Parameter für den Wolfram Automat 42.

- $k^p = 2^3 = 8$, $n_q = 5 \rightarrow \lambda = \frac{8-5}{8} = 0.375$

Lambda korreliert mit der Klasse. Somit lassen sich die Klassen ungefähr mit Hilfe dieses Parameters einteilen und Systemeigenschaften aus der Übergangsfunktion vorhersagen.

- Für $\lambda = 0$ sind keine Zellen aktiv
- Für $\lambda = 1$ sind alle Zellen aktiv
- Für $\lambda = 0.5$ ist die Hälfte der Zellen aktiv
- Zwischen 0.5 und 1 dreht sich das Verhalten ungefähr um, d.h. der Bereich $0 \leq \lambda \leq 0.5$ ist interessant

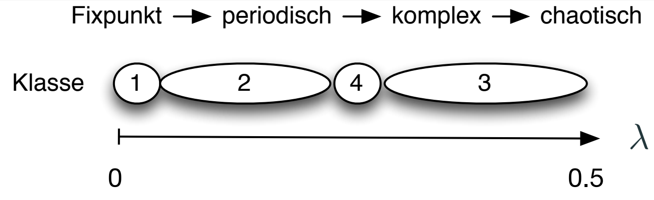


Abbildung 18: Beziehung Wolframs Klassen und Langton Parameter

Langton Parameter ist nur eine Abschätzung.

- Regel 18, binär 00010010, $\lambda = 0.25$, Klasse 3
- Regel 40, binär 00101000, $\lambda = 0.25$, Klasse 1

Abbildung 19: Regel 18 (links) und Regel 40 (rechts)

Zweidimensionale Automaten

- In den 1970 Jahren von John Horton Conway entwickelt
- Berühmtes Beispiel, dass emergente, selbst-organisierende Systeme aus einfachen Regeln entstehen können
- Hat die Berechnungskomplexität einer universellen Turingmaschine, d.h. alles, was algorithmisch berechenbar ist, kann mit diesem Zellulären Automaten berechnet werden

Abbildung 20: John Horton Conway (Thane Plambeck via Flicker CCA-2 Generic)

Beschreibung

- $n \times m$ Gitter, Randbedingung periodisch (Torso)
- Moore-Nachbarschaft (ohne die Zelle selbst, 8 Nachbarn)
- Sei $n_l(i, j, t)$ die Anzahl der aktiven Zellen von Zelle $z_{i,j}$
- Übergangsfunktion definiert durch

$$z_{i,j}(t+1) = \begin{cases} 1 & \text{für } z_{i,j}(t) = 0 \text{ und } n_l(i,j,t) = 3 \\ 1 & \text{für } z_{i,j}(t) = 1 \text{ und } n_l(i,j,t) \in \{2,3\} \\ 0 & \textit{sonst} \end{cases}$$

Analogie zum Leben

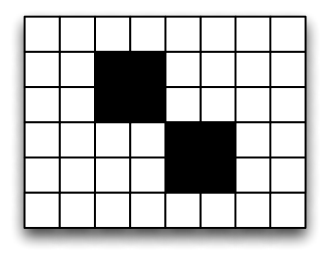
- Eine tote Zelle wird lebendig, wenn sie genau 3 lebende Nachbarn hat
- Eine lebende Zelle stirbt, wenn sie weniger als 2 (Einsamkeit) oder mehr als 3 (Überbevölkerung) lebende Nachbarn hat

Aufgabe

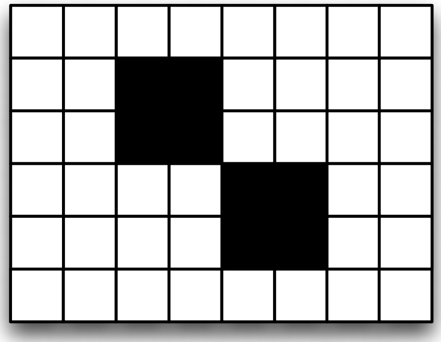
- Moore-Nachbarschaft
- Übergangsfunktion

$$z_{i,j}(t+1) = \begin{cases} 1 & \text{für } z_{i,j}(t) = 0 \text{ und } n_l(i,j,t) = 3 \\ 1 & \text{für } z_{i,j}(t) = 1 \text{ und } n_l(i,j,t) \in \{2,3\} \\ 0 & \textit{sonst} \end{cases}$$

Wie entwickelt sich folgender Anfangszustand ($t = 0$) über die Zeit ($t = 1, 2, \ldots$)?



Wie entwickelt sich folgender Anfangszustand ($t = 0$) über die Zeit ($t = 1, 2, \ldots$)?



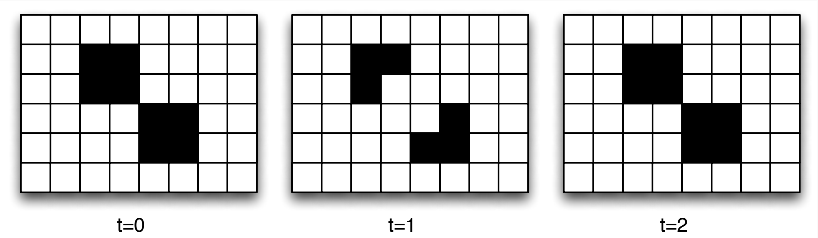


Abbildung 21: Oszillierendes Muster mit dem Namen ”beacon”

Eine Auswahl besonderer Muster im Game of Life

- Stabile Formen: ändern sich nicht über die Zeit
- Oszillatoren: zeigen periodisches Verhalten (z.B. der Beacon auf Folie 40)
- Gleiter: Muster, die sich über die Zeit durch den Raum bewegen
- Gleiterkanone: Muster, dass sich über die Zeit bewegt und dabei kleine Gleiter generiert

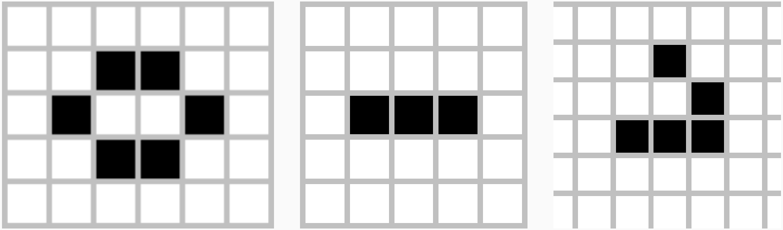


Abbildung 22: Beehive (stabil), Blinker (oszillierend), Gleiter (Public Domain via Wikimedia Commons)

Zum selber ausprobieren

- Sammlung von Lebensformen `http://pentadecathlon.com`
- Open Source Programm (Win, *nix, OSX, Android, iOS) `http://golly.sourceforge.net`

Anwendungsbeispiele

- 1971 von Thomas Schelling vorgestelltes sozialwissenschaftliches Modell [3]
- Beobachtung in den USA: trotz Bemühungen blieben Schulen, Wohngebiete, Stadtteile usw. in den USA getrennt nach Rassen und Einkommen
- Annahme: Menschen haben für andere ethnische Gruppen ein bestimmtest Toleranzlevel und fühlen sich nur wohl, wenn eine Mindestanzahl Personen gleicher Ethnie in ihrer Nähe wohnt.
- Wenn diese Mindestanzahl unterschritten wird, ziehen Menschen in eine Umgebung, in der sie sich wohler fühlen.
- Frage: Welcher Wert dieser Toleranzschwelle führt zu De-seggregation?

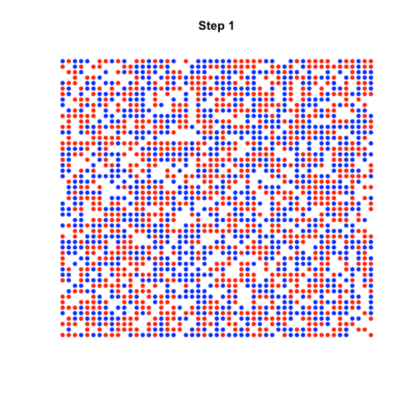
Schellings Seggregationsmodell

Beschreibung

- Moore-Nachbarschaft (8 Nachbarn)
- Zellzustände: 0 (frei),1 (Typ 1), 2 (Typ 2)
- Regel:
  - Übersteigt die Anzahl der Nachbarn, die einen anderen Zellzustand haben einen Schwellwert θ, dann bewegt sich die Zelle an einen „nahegelegenen“Ort, der mindestens θ Nachbarn desselben Zustandes hat
- Ein nahegelegener Ort wird über die erweiterte Moore-Nachbarschaft repräsentiert, Suche nach freien Orten mit aufsteigendem Nachbarschaftsradius
- Simulation asynchron (d.h. eine Zelle nach der anderen), solange, bis stabiler Zustand erreicht ist

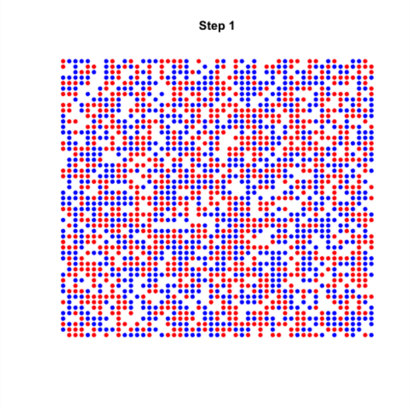
- Überraschender Fund: Bereits mit niedrigen Schwellwerten für θ trennen sich gemischte Nachbarschaften mit der Zeit
- Zum Beispiel mit 30% Schwellwert sind immer noch 70% aller Nachbarschaften getrennt, d.h. im Schnitt führen 2 von 8 Zellen in der Nachbarschaft, die einen anderen Typ haben zu globaler Seggregation

Simulation mit $\theta = 20\%$[1]



[1] PDF mit Adobe Acrobat Reader öffnen, um die Animation zu sehen. (Erstellt mit `http://rsnippets.blogspot.de/2012/04/animating-schellings-segregation-model.html`)

Simulation mit $\theta = 60\%$[2]



[2]PDF mit Adobe Acrobat Reader öffnen, um die Animation zu sehen. (Erstellt mit `http://rsnippets.blogspot.de/2012/04/animating-schellings-segregation-model.html`)

WATOR MODELL

- 1984 von Alexander K. Dewdney vorgeschlagen [1]
- Biologisches Modell, das verwendet wird, um Entwicklungen von Räuber-Beute-Populationen vorherzusagen
- Haie (Räuber) und Fische (Beute) leben in einer torusförmigen Wasserwelt (WATOR – water torus)
- Sowohl Haie als auch Fische reproduzieren sich

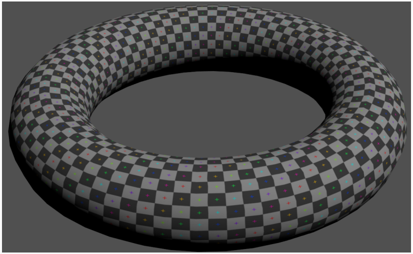


Abbildung 23: WATOR world (Iammajormartin via Wikimedia Commons, CC0, Public Domain dedication(

WATOR Modell

Beschreibung Teil 1

- Parameter
  - N_f: Anzahl Fische zum Startzeitpunkt
  - N_s: Anzahl Haie zum Startzeitpunkt
  - B_f: Anzahl Generationen bevor ein Fisch sich reproduzieren kann
  - B_s: Anzahl Generationen bevor ein Hai sich reproduzieren kann
  - S: Anzahl Generationen seit dem letzten Fressen bevor ein Hai den Hungerstod stirbt
- Zellzustände: 0 - Wasser, Fisch - 1, Hai - 2
- Initiale Verteilung zufällig

Beschreibung Teil 2

- Regeln
  - Fische schwimmen zufällig in freie Zellen nach N, O, S, W. Gibt es keine freie Zelle, bleiben sie wo sie sind.
  - Wenn in N, O, S, W Umgebung Fische sind, schwimmen Haie zu einem zufällig ausgewählten und fressen ihn. Der Fisch verschwindet aus der Welt. Wenn es keine Fische in der Umgebung gibt, schwimmen sie zufällig in eine freie Zelle. Haie schwimmen nicht in Zellen mit anderen Haien.
  - Wenn B_f bzw. B_s Generationen vorbei sind, können Haie/Fische brüten. Wenn eine Nachbarzelle frei ist, schwimmen sie dorthin, und hinterlassen einen Nachfolger in der gerade verlassenen Zelle. Für Eltern und Kinder wird der biologische Reproduktionszähler auf 0 gesetzt. Ist keine Zelle frei, passiert nichts. Ein Hai kann gleichzeitig einen benachbarten Fisch essen und sich vermehren.
  - Haie sterben, wenn sie S Generationen nicht gefressen haben. Nach jedem Fressen wird ihr Zähler zurück gesetzt.

Es gibt drei mögliche Ergebnisse der Simulation

1. Die Haie sterben aus, die Fische (über-)bevölkern das Wasser.
2. Die Fische sterben zuerst aus, damit auch die Haie an Hungersnot.
3. Gleichgewicht. Die Fische werden weniger, damit die Haie auch; die Fische können sich wieder reproduzieren, weil Platz ist; die Haie haben mehr Futter und vermehren sich usw.

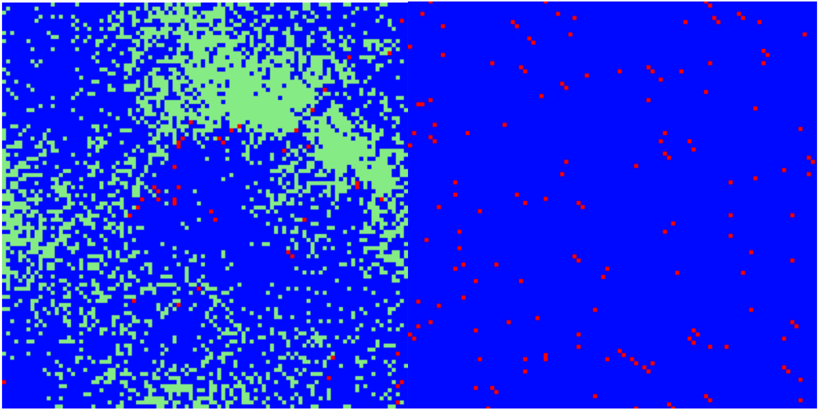


Abbildung 24: WATOR world, Haie rot, Fische blau. Haisterben (links), Fischsterben (rechts) (Bilder erstellt mit `https://www.cheesygames.com/wator/`)

Gelingt es Ihnen, ein stabiles Ökosystem zu schaffen?

`https://www.cheesygames.com/wator/`

Zusammenfassung

Zusammenfassung

- Zelluläre Automaten sind diskrete, räumlich-zeitliche Modelle
- Entwicklungsbedingungen beruhen auf Nachbarschaften (lokalen Regeln)
- Bereits Automaten mit sehr einfachen Regeln können komplexes Verhalten zeigen (z.B. Wolframs Regel 110 ist Turing-vollständig)
- Viele verschiedene Phänomen der realen Welt können mit Zellulären Automaten simuliert werden

Wichtige Konzepte

- Zellulärer Automat $CA = (L, S, N, f)$
    - Gitter
    - Zustände
    - Nachbarschaft (Moore, von-Neumann)
    - Übergangsfunktion
- Randbedingungen
- Wolframs Universum (1D)
- Game-of-Life (2D)
- Langton Parameter λ

Weiterführende Literatur

- Cellular Automata: A Discrete View of the World
  (Preprint PDF Version online verfügbar)
  Joel L. Schiff. *Cellular Automata: A Discrete View of the World*. Wiley, 2008. 252 S. ISBN: 978-0-470-16879-0. URL: http://psoup.math.wisc.edu/pub/Schiff_CAbook.pdf

Literatur

[1] Alexander K. Dewdney. “Dewdney, A.K., Sharks and fish wage an ecological war on the toroidal planet Wa-Tor”. In: *Scientific American* (Dez. 1984), S. 14–22. URL: http://home.cc.gatech.edu/biocs1/uploads/2/wator_dewdney.pdf.

[2] Addi Nüchel. *Chaosbox. Simulation zellulärer Automaten*. Heureka Klett, 1995. ISBN: 978-3121327607.

[3] Thomas C. Schelling. “Dynamic models of segregation”. In: *The Journal of Mathematical Sociology* 1.2 (1971), S. 143–186. DOI: 10.1080/0022250X.1971.9989794. URL: https://www.stat.berkeley.edu/~aldous/157/Papers/Schelling_Seg_Models.pdf.

[4] Joel L. Schiff. *Cellular Automata: A Discrete View of the World*. Wiley, 2008. 252 S. ISBN: 978-0-470-16879-0. URL: http://psoup.math.wisc.edu/pub/Schiff_CAbook.pdf.