

Darstellung von Differentialgleichungen mit einem zellulären Automaten

Detlev Ziereisen, Florian Lüthi, i10b

28. April 2012

1 Einleitung

2 Zelluläre Automaten

Ein zellulärer Automat ist eine regelmäßige Annordnung von Zellen. Jede Zelle kann eine endliche Zahl von Werten / Zuständen annehmen und hat eine begrenzte Zahl von Nachbarzellen, die sie beeinflussen können. Das Muster des gesamten zellulären Automaten ändert sich in einzelnen Schritten, die durch eine Reihe von Übergangsregeln bestimmt werden, die für alle Zellen gelten.[?]

Also:

Definition 1 (Zellulärer Automat). Ein zellulärer Automat ist durch folgende Eigenschaften festgelegt:

- einen Zellularraum R ,
- eine endliche Nachbarschaft N , wobei $\forall r \in R (N_r \subset R)$,
- eine Zustandsmenge Q ,
- eine Überföhrungsfunktion $\delta : Q^{|N|+1} \mapsto Q$.

Die Zustandsübergänge erfolgen für alle Zellen nach derselben Überföhrungsfunktion und gleichzeitig. Die Zellzustände können wie die Zeitschritte diskret sein. [?]

Bemerkung. Aus dieser Definition folgt unmittelbar, dass der neue Zustand einer Zelle nur vom momentanen Zustand dieser Zelle sowie den Zuständen der Nachbarzellen abhängig sein kann.

Für zweidimensional organisierte R sind zwei Arten von Nachbarschaft üblich:

Von-Neumann-Nachbarschaft Die Nachbarschaft besteht jeweils aus den vier geographisch nächsten Nachbarzellen: $N_{i,j} = \{R_{i,j-1}, R_{i,j+1}, R_{i-1,j}, R_{i+1,j}\}$

Moore-Nachbarschaft Die Nachbarschaft besteht jeweils aus allen Zellen der Von-Neumann-Nachbarschaft sowie zusätzlich der diagonalen Nachbarzellen:

$$N_{i,j} = \{R_{i,j-1}, R_{i,j+1}, R_{i-1,j}, R_{i+1,j}, R_{i-1,j-1}, R_{i-1,j+1}, R_{i+1,j-1}, R_{i+1,j+1}\} \text{ [?]}$$

Beispiel (Wolfram's eindimensionales Universum). Stephen Wolfram¹ definiert in [?] und in etlichen Arbeiten aus der Mitte der 1980er-Jahre einen parametrierbaren zellulären Automaten, der nur aus einer einzigen Raumdimension besteht[?]. In seiner einfachsten Ausprägung ist die Nachbarschaft N_i definiert als $\{R_{i-1}, R_{i+1}\}$, und jede Zelle kann genau zwei Zustände annehmen (tot und lebendig bzw. 0 und 1). Damit ist

$$\delta : \{0, 1\}^{|\{R_{i-1}, R_{i+1}\}|+1} \mapsto \{0, 1\} = \{0, 1\}^3 \mapsto \{0, 1\},$$

ergo existieren 8 mögliche Zustandsänderungen. Ein Beispiel:

alte $R_{i-1}R_iR_{i+1}$	000	001	010	011	100	101	110	111
neues R_i	0	1	0	1	1	0	1	0

Tabelle 1: Wolfram-Konfugation 110

Man stellt nun fest, dass bei 8 möglichen Zustandsänderungen für die Anzahl der möglichen Konfigurationen gilt:

$$|K| = |Q|^{|\text{dom}(\delta)|} = 2^8 = 256.$$

Jeder dieser Konfigurationen kann eine natürliche Zahl $\{0, 1, 2, \dots, 255\}$ zugeordnet werden, indem die neuen R_i wie oben tabelliert und pro Zeile als binäre Zahl aufgefasst werden [?]:

000	001	010	011	100	101	110	111	Zahl
0	0	0	0	0	0	0	0	$00000000_b = 0_d$
0	0	0	0	0	0	0	1	$00000001_b = 1_d$
0	0	0	0	0	0	1	0	$00000010_b = 2_d$
\vdots								
0	1	0	1	1	0	1	0	$01011010_b = 110_d$
\vdots								
1	1	1	1	1	1	1	1	$11111111_b = 255_d$

Tabelle 2: Alle Wolfram-Konfigurationen

Dadurch ist es möglich, sämtliche eindimensionalen Wolfram-Universen durch eine einzige Zahl zu identifizieren.

¹Stephen Wolfram (* 29. August 1959), britischer Physiker und Mathematiker, Schöpfer der Software Mathematica sowie der Suchmaschine Wolfram Alpha [?]

Als besonders spannend hat sich die in Tabelle ?? dargestellte Konfiguration 110 erwiesen, weil sie die Eigenschaft hat, ein Turing-vollständiges System zu sein [?, ?]. Dadurch ist sie die Konfiguration einer universellen Turingmaschine, die mit nur 2 Zuständen und 5 Symbolen umgesetzt werden kann – somit hat die Wolfram-Konfiguration 110 als Turingmaschine einen Umfang von $2 \cdot 5 = 10$ und zählt damit zu den kleinsten bis dato bekannten Turingmaschinen [?].

Beispiel (Game of Life). Das von Conway² 1970 entworfene Game of Life ist eine bis heute populäre Umsetzung der Automatentheorie und insbesondere der Idee der zellulären Automaten [?].

Der ursprüngliche Entwurf befindet sich in einem zweidimensionalen R unter Verwendung der Moore-Nachbarschaft. Die Zellen können zwei mögliche Zustände $\{q_{\text{lebend}}, q_{\text{tot}}\}$ annehmen, und die Übergangsfunktion ist definiert als:

$$\delta(r, N) = \begin{cases} q_{\text{lebend}} & (r = q_{\text{tot}} \wedge \varsigma(q_{\text{lebend}}, N) = 3) \\ q_{\text{tot}} & (r = q_{\text{lebend}} \wedge \varsigma(q_{\text{lebend}}, N) < 2) \\ q_{\text{lebend}} & (r = q_{\text{lebend}} \wedge 2 \leq \varsigma(q_{\text{lebend}}, N) \leq 3) \\ q_{\text{tot}} & (r = q_{\text{lebend}} \wedge \varsigma(q_{\text{lebend}}, N) > 3) \\ q_{\text{tot}} & (\text{sonst}) \end{cases}$$

unter Zuhilfenahme der Statuszählfunktion

$$\varsigma(q, N) = \sum_{i=1}^8 \begin{cases} 1 & (N_i = q) \\ 0 & (N_i \neq q) \end{cases}$$

3 Vom zellulären Automaten zur Differentialgleichung

Die Berechnung und Darstellung physikalischer Begebenheiten (allgemein ausgedrückt durch folgende partielle Differentialgleichung, mit $u : u(\vec{x}, t)$)

$$k_n \frac{\partial^n u}{\partial t^n} + k_{n-1} \frac{\partial^{n-1} u}{\partial t^{n-1}} + \dots + k_1 \frac{\partial u}{\partial t} + k_0 = \frac{\partial^n u}{\partial \vec{x}^n} + \frac{\partial^{n-1} u}{\partial \vec{x}^{n-1}} + \dots + \frac{\partial u}{\partial \vec{x}}$$

mit zellulären Automaten kann durchgeführt werden, indem folgendes getan wird:

- R entspricht einer sinnvollen (groben) Diskretisierung der örtlichen Variablen \vec{x} in einer, zwei oder drei Dimensionen
- Jede Zelle in R ist ein Tupel (Q, D) mit Q als einer Menge von berechnungsfernen Zustandsinformationen und $D = \left(u, \frac{\partial u}{\partial \vec{x}}, \frac{\partial^2 u}{\partial \vec{x}^2}, \dots, \frac{\partial^n u}{\partial \vec{x}^n}\right) \in \mathbb{R}^n$
- Eine neue Generation entspricht jeweils der fortgelaufenen Zeit ∂t , welche sehr fein diskretisiert werden muss
- In der Übergangsfunktion δ steckt die eigentliche Differentialgleichung. In der Regel verändert sie nur die Elemente von D .

²John Horton Conway (* 26. Dezember 1937), englischer Mathematiker[?]

- 4 Implementierte Differentialgleichungen
- 5 Numerische Verfahren
- 6 Bemerkungen zur Implementation
- 7 Schlussfolgerung