

EIN MULTI-SPEED-PERSONENSTROMSIMULATOR AUF BASIS EINES ZELLULAREN AUTOMATEN

Tobias Liepert¹, André Borrmann¹, Wolfram Klein²

¹Computation in Engineering, Technische Universität München

borrmann@bv.tum.de, liepert@mytum.de

²Corporate Technology, Siemens AG, München

wolfram.klein@siemens.com

Kurzfassung: Ziel der Entwicklung moderner Personenstromsimulatoren ist, die reale Welt möglichst genau abzubilden und gleichzeitig die Simulationslaufzeit zu minimieren. Der Multi-Speed-Ansatz als wesentlicher Kernpunkt neuer Simulatoren ermöglicht es, trotz diskreter Simulationsgrößen das reale Bewegungsverhalten unterschiedlicher Personen mit Hilfe variabler Geschwindigkeitsklassen abzubilden. Voraussetzung dafür ist eine sinnvoll Wahl der diskreten Simulationsgrößen, die dem Simulator zu Grunde liegen. Das Paper beschäftigt sich mit der analytischen Beschreibung dieser diskreten Simulationsgrößen und ihrer quantitativen Kalibrierung. Dabei stellt die Definition der Zeitgrößen, die einen Bezug zwischen Simulator und Realwelt ermöglichen, und so die Grundlagen für eine Interpretation der Simulationsergebnisse bilden, einen Schwerpunkt der Ausführungen dar.

1 Grundlagen

Prinzipiell ist zwischen mikroskopischen und makroskopischen Fußgängermodellen zu unterscheiden. Im mikroskopischen Modell werden die Personen mittels eigenständiger Objekte abgebildet. Im makroskopischen Modell werden Fußgängerströme betrachtet und die Auswertung erfolgt an Hand strömungsmechanischer Berechnungen. Im Rahmen dieses Papers wurde ein mikroskopisches Fußgängermodell gewählt, das auf dem Prinzip von zellularen Automaten [2-6] beruht.

Ein zellulärer Automat ist durch folgende Größen festgelegt:

- ein Raum R (Zellraum)
- eine Nachbarschaftsrelation

- eine endliche Zustandsmenge Q
- eine lokale Überföhrungsfunktion

2 Zellularer Automat

2.1 Zellraum

Ein reguläres Gitter ist die Grundlage eines zellularen Automaten. Dreieck-, Rechteck- und Hexagongitter bilden reguläre Gittergeometrien und eignen sich so hervorragend für die Gebietsdiskretisierung. Fünfeck- oder Achteckgitter hingegen sind in Folge ihrer Diskontinuitäten für die Gebietsdiskretisierung ungeeignet (Abb. 2.1).

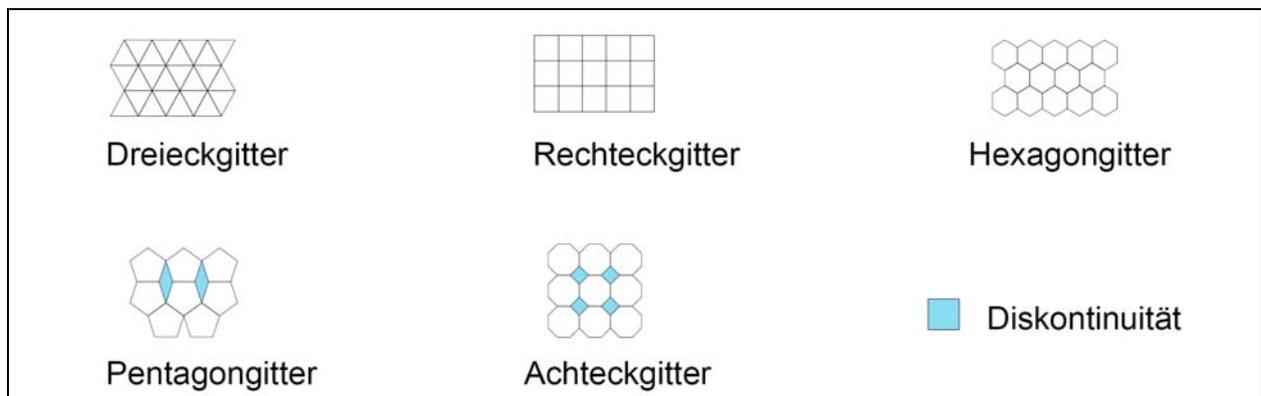


Abb. 2.1: Gittertypen

Der Rand der Gebietsdiskretisierung wird allseits geschlossen ausgebildet, auf besondere Übergangsbedingungen kann somit verzichtet werden. Die Größe des Gitters wird durch das Staukriterium festgelegt, welches beschreibt, wie viele Personen sich auf einem Quadratmeter bei eintretendem Stau aufhalten.

2.2 Nachbarschaftrelation

Die Beziehung von Zellen, die eine gemeinsame Kante besitzen, ist der erster möglicher Ansatz und wird als **Von-Neuman-Nachbarschaft** bezeichnet. Der zweite Ansatz ist eine Erweiterung des Ersten und wird als **Moore'sche Nachbarschaft** bezeichnet, dabei sind nicht nur alle Zellen mit einer gemeinsamen Kante Nachbarn, sondern auch Zellen mit einer gemeinsamen Ecke (siehe [2]).

Das Hexagongitter besitzt unabhängig von der betrachteten Nachbarschaft eine eindeutige Nachbarschaftsrelation, zudem ist der Abstand der Zellmittelpunkte konstant [2]. Diese Kriterien sind ausschlaggebend für die Wahl des Hexagongitters als Zellraumdiskretisierung (Abb. 2.2).



Abb. 2.2: Nachbarschaftsrelationen

2.3 Zustandsmenge

Jeder Zelle des zellularen Automaten muss ein eindeutiger Zustand zugeordnet werden. Die Zustandsmenge fasst alle Zustände zusammen und besteht aus folgenden Komponenten:

- Die Zelle ist unbesetzt.
- Auf der Zelle befindet sich eine Person.
- Die Zelle ist mit einem Hindernisobjekt besetzt.
- Die Zelle ist mit einer Quelle besetzt.
- Die Zelle ist mit einem Ziel besetzt.

2.4 Übergangsfunktion

Der Zustandsübergang muss für alle Zellen nach der selben Übergangsfunktion erfolgen. Mit dem Ziel Personenbewegung abzubilden, ist die Hauptaufgabe der Übergangsfunktion die Zustandsänderung von einer unbesetzten Zelle, zu einer durch eine Person besetzten Zelle und umgekehrt. Ferner, dass im direkten Umkreis von Quellen Zellen durch Personen besetzt und beim Einlauf in das Ziel Personen gelöscht werden.

Der Multi-Speed-Ansatz wird umgesetzt, indem jeder Person eine diskrete Geschwindigkeit zugeordnet ist, mit der sie unter Voraussetzung bestimmter Kriterien (siehe Pseudocode 1) ziehen darf.

```

For i = 1 to  $n_{mFC}$ 
  For All_Pedestrians p
    If p.habitualSpeed >= i
      p.move
    next p
  For All_Sources generate pedestrians
  For All_Targets delete pedestrians
next i

```

} Bewegungsschritt
 } Iterationsschritt

Pseudocode 1: programminterner Ablauf zur Personenbewegung

Die äußere Schleife, ein Iterationsschritt, läuft über alle Zellgeschwindigkeitsklassen und die innere Schleife über alle Personen. Die Reihenfolge der Personen wird nicht explizit festgelegt, sondern entspricht der Reihenfolge des Array, indem alle Personen gespeichert sind. Die if-Abfrage in der inneren Schleife in Pseudocode 1 verhindert, dass eine Person mit einer Geschwindigkeit von x beim $(x+1)$ -Zeitschritt sich bewegen darf.

3 Geschwindigkeit

3.1 Allgemeines

Kennzeichnend für die Bewegung eines Fußgängers ist die Wahl seiner individuellen Wunschgeschwindigkeit. Diese wird bestimmt von persönlichen Faktoren wie etwa dem Alter und dem Geschlecht, aber auch Einflüsse der Umwelt spielen eine wesentliche Rolle.

Einflussfaktoren:

- persönliche Faktoren (Alter, Geschlecht, Aktivität)
- gewöhnliche Einflüsse der Umwelt (Fußgängerichte, Tageszeit, Jahreszeit, Wetter)
- außergewöhnliche Einflüsse (Notfallsituation, Überfall, Feuer)

WEIDMANN [1] konnte trotz der Komplexität der Einflüsse bei empirischen Versuchen einen Zusammenhang zwischen der Personenstromdichte und der Geschwindigkeitsverteilung herstellen. Die Auswertung (siehe Abb. 3.1) zeigt die Versuchsergebnisse, als genäherte Gauß-verteilte Glockenkurven.

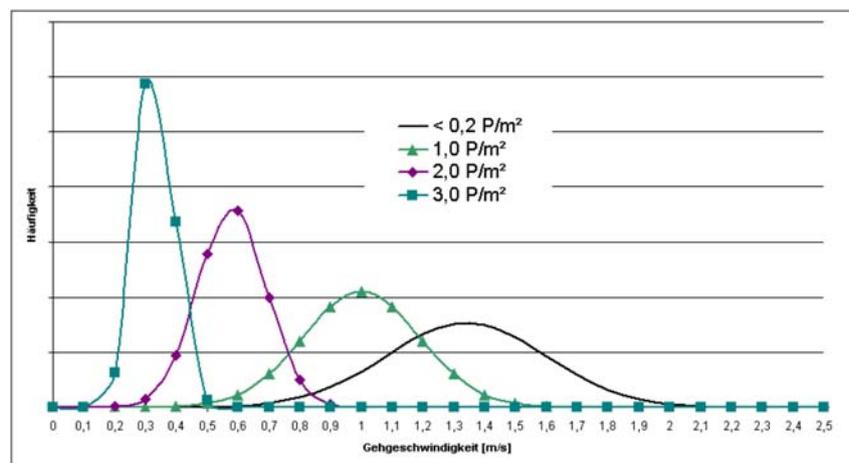


Abb. 3.1: Fußgängerichte und Fußgängergeschwindigkeit nach Weidmann[1]

Die oben dargestellten Geschwindigkeitsverteilungen gelten nur für einen leicht bekleideten Mitteleuropäer mit leichtem Handgepäck (Tasche).

3.2 Zellgeschwindigkeit / Realgeschwindigkeit

Die Zellgeschwindigkeit v_c (*cell velocity*) beschreibt, wie viele Zellen während eines Iterationsschritts zurückgelegt werden, die Einheit ist [Zellen / Iterationsschritt]. Die Realgeschwindigkeit v_r (*real velocity*) entspricht der physikalischen Größe, der Geschwindigkeit, gemessen in [m/s].

3.3 Diskretisierung der Geschwindigkeit

Die Geschwindigkeitsdiskretisierung beschreibt den Vorgang, indem die Realgeschwindigkeiten in Klassen zusammengefasst und die Klassenmittelwerte anschließend als zugehörige Zellgeschwindigkeit gespeichert werden. Die verwendete Verteilungsfunktion ist, die von WEIDMANN [1] aufgestellte Gauß-Verteilung bei 0,2 Pers./sec.

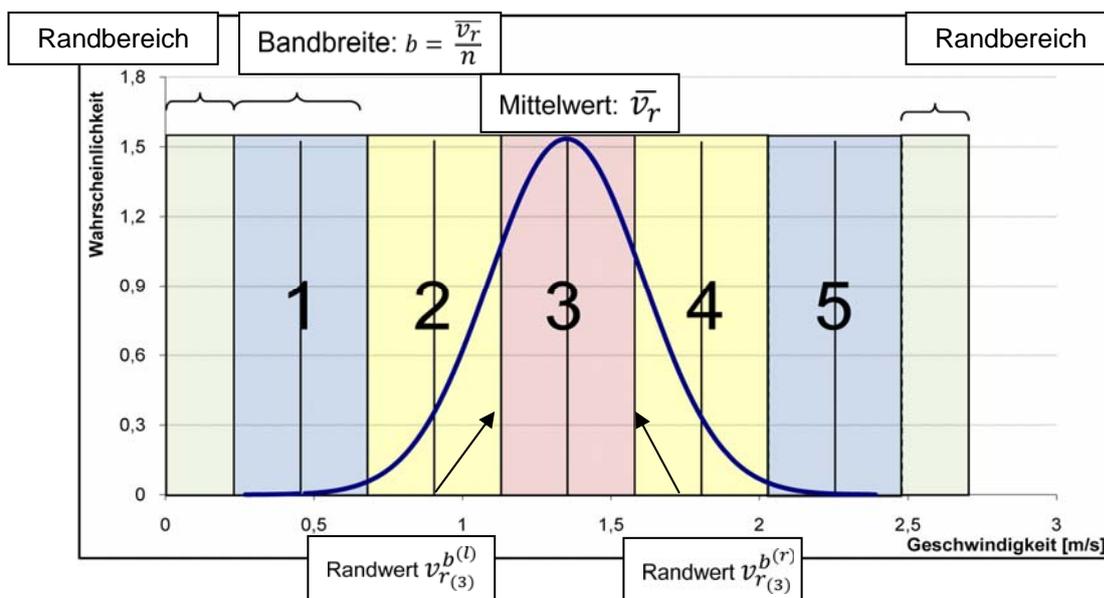


Abb. 3.2: Geschwindigkeitsklassenbildung

3.3.1 Klassenbildung der Geschwindigkeitsklassen

Im ersten Schritt wird die mittlere Zellgeschwindigkeit \bar{v}_c dem Mittelwert der standard-normalverteilten Realgeschwindigkeit \bar{v}_r zugeordnet.

$$\bar{v}_r \rightarrow \bar{v}_c$$

Ausgehend von dieser Mitte werden nach links und recht im Abstand (x) neue Zellgeschwindigkeitsklassen definiert, der Klassenmittelwert der Zellgeschwindigkeit ergibt sich zu

$$v_{c(\bar{v}_c+x)} = (\bar{v}_c + x) \text{ für } x \in [(-\bar{v}_c + 1); (\bar{v}_c - 1)] \wedge x \in \mathbb{N}$$

$$x = (\bar{v}_c - v_{c(\bar{v}_c+x)})$$

Die dazugehörigen Realgeschwindigkeiten bestimmen sich aus

$$v_{r(\bar{v}_c+x)} = (\bar{v}_c + x) \times \frac{\bar{v}_r}{\bar{v}_c} \text{ für } x \in [(-\bar{v}_c + 1); (\bar{v}_c - 1)] \wedge x \in \mathbb{N}$$

Die realen Randwerte (links l, rechts r) der Geschwindigkeitsklassen sind

$$v_{r(\bar{v}_c+x)}^{b(l,r)} = v_{r(\bar{v}_c+x)} \pm \frac{b}{2} = (\bar{v}_c + x) \times \frac{\bar{v}_r}{\bar{v}_c} \pm \frac{b}{2} = \left(\bar{v}_c + x \pm \frac{1}{2} \right) \times b$$

Die Bandbreite b der Geschwindigkeitsklassen ergibt sich aus

$$b = \frac{\bar{v}_r}{\bar{v}_c}$$

Die Umrechnung von Zellgeschwindigkeiten in Realgeschwindigkeiten ergibt sich wie folgt

$$b \times \bar{v}_c = \bar{v}_r \quad \text{allgemein} \quad b \times v_c = v_r$$

Die Anzahl der Diskretisierungsschritte ist gleich der Anzahl der Geschwindigkeitsklassen, dies entspricht der maximalen Diskretisierungsauflösung.

$$n_{\text{maxVelocityClasses}} = n_{mVC} = (2 \times \bar{v}_c) - 1$$

3.3.2 Diskretisierungsvorgang

Die eigentliche Diskretisierung benötigt zwei Komponenten. Zum Einen Personen mit einer zufälligen, realen Gewohnheitsgeschwindigkeit v_r^P und zum Anderen die Geschwindigkeitsklassen mit ihren jeweiligen Randwerten. Folgender Zusammenhang muss für ein $x \in [(-\bar{v}_c + 1); (\bar{v}_c - 1)] \wedge x \in \mathbb{N}$ gelten

$$v_{r(n+x)}^{b(l)} \leq v_r^P \leq v_{r(n+x)}^{b(r)} \implies v_c^P = v_{c(n+x)}$$

3.3.3 Fehlerberechnung

Die Fehlerberechnung ergab, bei einer Diskretisierung mit $n = 2$ einen Fehler von nur 0,000111 (jede 1000-ste Person), dies ist ein akzeptables Maß.

4 Zeitgrößen

4.1 Reale Zeitgrößen

Als maßgebende Zeitgrößen werden im Simulator zwei Begrifflichkeiten definiert:

- **Bewegungsschritt:** Dauer einer Zell-Bewegung
- **Iterationsschritt:** Dauer mehrerer Zell-Bewegungen, entsprechend der Zellgeschwindigkeit

Der Iterationsschritt besteht immer aus mehreren Bewegungsschritten, die Iterationsschrittdauer entspricht somit einem Vielfachen der Bewegungsschrittdauer. Für die reale Bewegungsschrittdauer gilt:

$${}^P\Delta t_m = \frac{s}{{}^P v_r}$$

Der Weg s ist der Abstand der Zellmittelpunkte, die Geschwindigkeit v_r entspricht der mittleren Realgeschwindigkeit, der zugeordneten Geschwindigkeitsklasse.

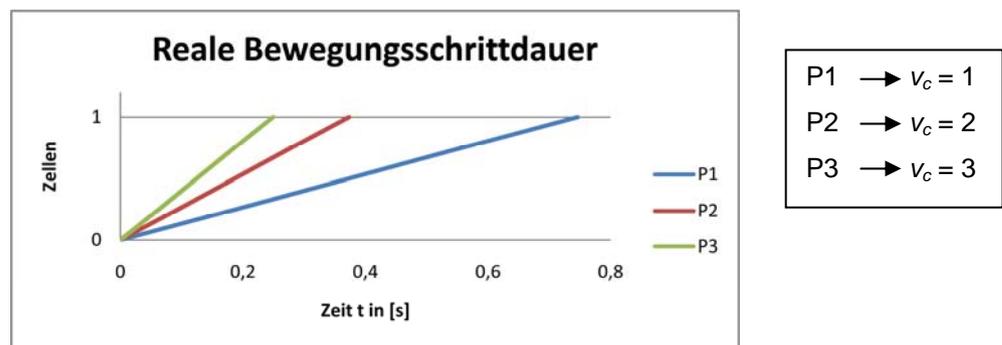


Abb. 4.1: Reale Bewegungsschrittdauer

Die reale Iterationsschrittdauer errechnet sich zu:

$$t_i = {}^P v_c \times {}^P \Delta t_m = \frac{\bar{v}_c}{\bar{v}_r} \times s = konst.$$

Dabei entspricht ${}^P v_c$ der zugewiesenen Zellgeschwindigkeit.

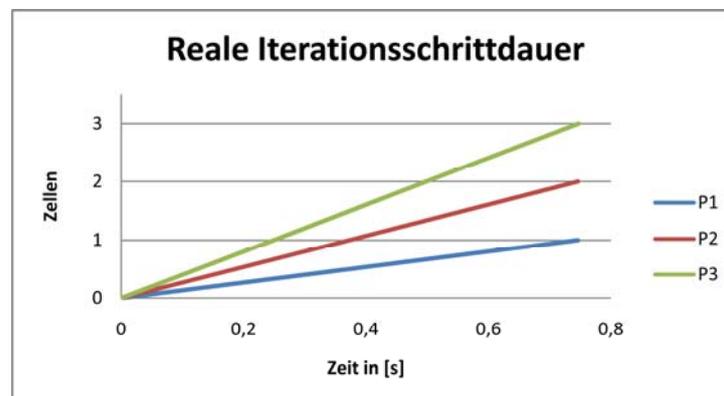


Abb. 4.2: Reale Iterationsschrittdauer

4.2 Simulationszeitgrößen

Die Überföhrungsfunktion (siehe 2.4) wurde so festgelegt, dass die kleinste Bewegungsschrittdauer die Schrittweite der äußeren For-Schleife (siehe Pseudocode 1) ist. Die kleinste Bewegungsschrittdauer errechnet sich zu:

$$t_{m,min} = \frac{t_i}{n_{mVC}} = \frac{\bar{v}_c \times s}{\bar{v}_r \times n_{mVC}} = \frac{\bar{v}_c \times s}{\bar{v}_r \times (2\bar{v}_c - 1)} = \frac{\bar{v}_c \times s}{2\bar{v}_r\bar{v}_c - \bar{v}_r}$$

Versucht man den realen Vorgang exakt abzubilden, so ergeben sich z.B. bei Person P1 und Person P2 (siehe Abb. 4.2) Zwischenzustände (Personen stehen zwischen zwei Zellen), die im zellularen Automaten nicht abgebildet werden können. In der Simulation ist es folglich nicht möglich, variierende kontinuierliche Bewegungsvorgänge exakt abzubilden.

Der entworfene Multi-Speed-Ansatz nähert die Real-Bewegung an, indem die Personen von Zelle zu Zelle (siehe Abb. 4.3) springen. Die Zellgeschwindigkeit beschreibt dabei, wie oft eine Person während eines Iterationsschritts springen darf. Die horizontalen Abschnitte in Abb. 4.3 stellen die Wartezeit der Personen dar. Am Ende jedes Bewegungsschrittintervalls dürfen all diejenigen Personen springen, die die notwendigen Voraussetzungen dafür erfüllen (siehe Pseudocode 1), die übrigen Personen müssen stehenbleiben (warten).

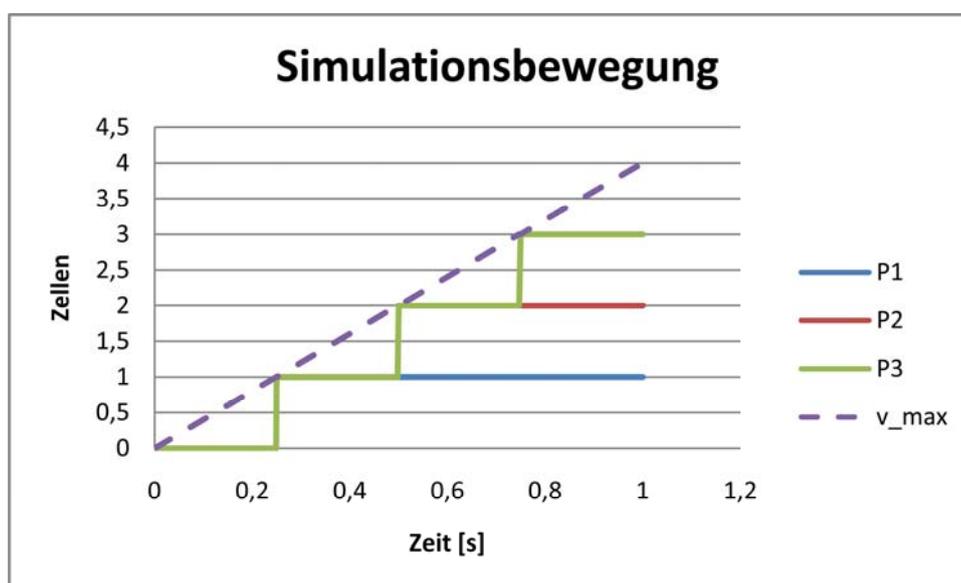


Abb. 4.3: Simulationsbewegung eines Bewegungsschritts

Die Gradienten der Treppen entspricht der Geschwindigkeit der Personen und errechnet sich wie folgt:

$$v = \frac{ds}{dt} = \frac{\Delta s}{\Delta t}$$

Die Gradiente der Treppen stellt die maximale Geschwindigkeit dar. In Abb. 4.4 ist der ungestörte Bewegungsablauf einer Person mit $v_c = 1$ und mittlerer Zellgeschwindigkeit $n = 2$ dargestellt.

Abb. 4.3 und Abb. 4.4 verdeutlichen die Funktionsweise des Multi-Speed-Ansatzes. Alle Personen bewegen sich während eines Bewegungsschritts mit gleicher Geschwindigkeit. Die Geschwindigkeitsunterschiede kommen zu Stande, indem die einzelnen Personen unterschiedlich oft springen dürfen.

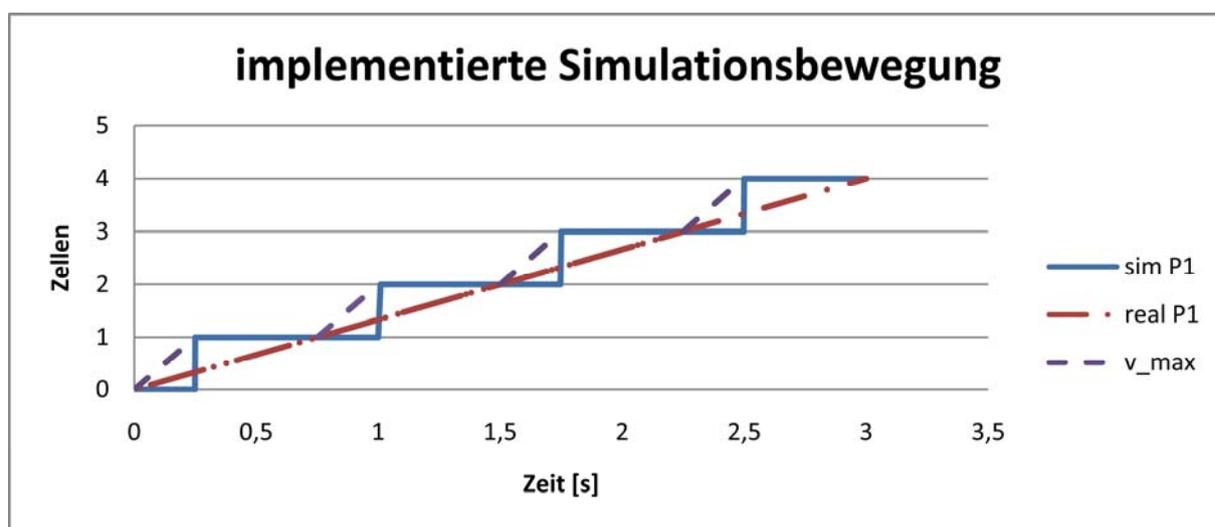


Abb. 4.4: Simulationsbewegung (mehrere Iterationsschritte)

Literatur

- [1] Weidmann, U. (1992). Transporttechnik der Fußgänger. Schriftenreihe des IVT, 90.
- [2] Kinkeldey, C. (2003). Fußgängersimulation auf der Basis sechseckiger zellularer Automaten. Studienarbeit. Institut für Bauinformatik. Hannover.
- [3] Blue, V.; Embrechts, M. & Adler, J. (1997) "Cellular automata modeling of pedestrian movements." IEEE International Conference on Systems, Man, and Cybernetics., 3 2320-2323
- [4] Burstedde, C.; Klauck, K.; Schadschneider, A. & Zittartz, J. (2001) "Simulation of Pedestrian Dynamics using a 2-dimensional Cellular Automaton." Phys. Rev. A, 295 507–525
- [5] Yue, H.; Hao, H.; Chen, X. & Shao, C. (2007) "Simulation of pedestrian flow on square lattice based on cellular automata model." Physica A: Statistical Mechanics and its Applications, 384 (2) 567-588
- [6] Was, J. (2005) "Cellular Automata Model of Pedestrian Dynamics for Normal and Evacuation Conditions." ISDA '05: Proc. of the 5th Int. Conf. on Intelligent Systems Design and Applications, IEEE Computer Society, 154-159