Automaton model with variable cell size for the simulation of pedestrian flow

Minjie Chen Günter Bärwolff Hartmut Schwandt

minjie.chen@math.tu-berlin.de baerwolf@math.tu-berlin.de schwandt@math.tu-berlin.de

Technische Universität Berlin, Fakultät II, Institut für Mathematik,
Sekr. MA 6-4, Straße des 17. Juni 135, 10623 Berlin, Germany

March 31, 2008

Abstract: The simulation of pedestrian flow is an important issue in many modern traffic systems and environments. The planning and control of pedestrian flow is crucial for large buildings, shopping malls, airports, railway stations, cruise ships and ferries, manifestation places, sport stadiums etc., in order that optimal traffic throughput for daily usage can already be guaranteed at the time of construction. Even evacuation scenarios can be considered in this context. Common simulation approaches are either discrete or continuous. While continuous models, which rely on mathematical models originally developed for fluids and gases, are better suited for the modelling of somehow global phenomena, individual aspects are given more consideration in the discrete models. Many discrete models for pedestrian flow are based on cellular automata, however, traditional cellular automaton models work with a fixed cell size. Since at any given time, the states of the cells are always well-defined by the automaton's transition rules, the current simulation object positioned in a grid cell can be considered as the cell state accordingly. In view of the original definition of the cellular automaton, we can understand the cell size as the minimal two-dimensional space the simulation objects (e. g. pedestrians) reserve for themselves exclusively. In the current paper, we present a two-dimensional automaton model to simulate pedestrian flow in which the cell size can be dynamically modified in a virtual manner. It can be easily observed in real world scenarios that the aforesaid minimal but exclusive space a pedestrian takes may be "elastic" (variable), especially when pedestrian density and the discrepancy of the flow-in and flow-out rates of the system are given sufficient consideration. Thus, this virtual flexibility of the cell size is favourable and necessary in the simulation of pedestrian flow where the minimal space required by a pedestrian is not confined to be constant. The virtual cell size in our model is constructed to be dependent on the actual pedestrian density. In the concluding section, we discuss the possibility of extending this model for the application of anisotropic neighbourhoods. The present work represents a continuation and extension of [1] and [3]. By introducing the concept of variable cell size and two different implementations for it, we significantly extend the applicability of cellular automaton models for the discrete simulation of pedestrian flow.

Keywords: cellular automaton, pedestrian flow, simulation, variable cell size

1 Introduction

Mathematical description and modelling of pedestrian flow has been intensively investigated in the last fifteen years (see e. g. [6], [8] and [16]). Many microscopic models have been proposed in which the pedestrians are considered as individuals. The concept of cellular automaton is appropriate for the handling of pedestrians in a heterogeneous way. In an early stage, [12] and [15] suggested a one-dimensional cellular automaton model for simulation of traffic flow in a closed, circle-shaped system. This model was later extended by [4] to function as a multi-lane traffic system.

An overview of pedestrian flow as an evacuation problem can be found in [5] where network/graph representations and cellular automaton models were discussed. The general pedestrian flow simulation is normally realized on a two-dimensional automaton model ([2], [9], [10], [11], [13] and [14]). These models share the characteristic that the step length of a pedestrian in a single simulation step must not be larger than one grid cell length (width). This is usually written as $v_{\max} = 1$. Like the majority of the authors, we call pedestrians *particle*s in the sequel.

From the perspective of mathematical analysis, the change of the particle velocity vector in the simulation can be understood as the result of the interactions among the particles and other simulation objects (e. g. interior walls). A complex mathematical formulation of the in-

teractions among particles and walls was proposed in [7]. Taking into consideration similar interactions among the particles, the interior walls and other extensible simulation objects, the two-dimensional automaton model of [1] and [3] offered a new ansatz to study the case of varied step lengths, i. e. $v_{\max} > 1$. Step conflicts in this model are solved using Bresenham's algorithm for line rastering.

In all existing two-dimensional cellular automaton models, choosing a suitable cell size, in our opinion, should not be of less importance—since the particles are considered to be heterogeneous—than conflict management. However, not much attention has been paid to this topic. In the current work, we would like to introduce a simple model to achieve the effect of modifiable cell size of a cellular automaton for pedestrian flow simulation.

2 Model

The geometry of the simulation system is as usual defined on a two-dimensional regular grid Ω of quadratic cells. Since the cells are referenced by their indices, we consider the grid Ω as a set of indices. In case of a rectangular region, we have a grid $\Omega = \{1, \ldots, n_{\text{row}}\} \times \{1, \ldots, n_{\text{col}}\}$ of $n_{\text{row}} \cdot n_{\text{col}}$ cells, where n_{row} and n_{col} denote the numbers of rows and columns respectively. Therefore, in this case Ω is a set of two-dimensional indices. Each cell is represented by an index pair (r, c), for $1 \le r \le n_{\text{row}}$, $1 \le c \le n_{\text{col}}$.

The state of an automaton cell is defined by the automaton's transition rules. The cells' states can be used to represent the simulation objects in these cells. Since the cell state is always unambiguously defined, there can exist at most one simulation object in each cell.[1] In most cases, this simulation object refers to the particle (i. e. the pedestrian). Hence, the cell size is comparable to the *minimal* two-dimensional space which a pedestrian in a real-world scenario reserves for himself exclusively. A typical cell size (suggested e. g. by [2]) is $0.4\,\text{m} \cdot 0.4\,\text{m}$. [18] proposed a model for pedestrian crossing dynamics with step length larger than 1. Besides, a particle is allowed to take one of the three moving directions (forward, left and right). Forward movement is given a step length twice as much as a sideward movement. In this model, a cell has a size of $0.48\,\text{m} \cdot 0.24\,\text{m}$, and a particle occupies exactly two cells (with their long edges attached to each other).

Concerning the biological and physiological aspect of human beings, the minimal plane space that a human needs, ranges from $0.09\,\text{m}^2$ to $0.16\,\text{m}^2$ approximately. This is an extreme case. In a normal situation, people as social creatures prefer a much larger space. On the other hand, as mentioned above, a cell can be occupied by at most one object (e. g. particle), it is feasible to choose a cell size of $0.3\,\text{m} \cdot 0.3\,\text{m}$ on Ω. This size is the minimal size a real pedestrian needs. However, we shall notice that often in an area composed of multiple cells—because real human beings prefer keeping a larger distance from each other—there should be at most one particle. We call such an area a *virtual* cell. Virtual cells can have different sizes, their minimum is the size of the real grid cell.

In our model, we choose 0.3 m as length (width) of the (basic) cell and we consider the Moore neighbourhood with index (also called distance) $\imath = 1$, i. e. the two-dimensional index set of relative positions $\{-1, 0, 1\} \times \{-1, 0, 1\} \setminus (0, 0)$.[2] $(0, 0)$ refers to the index of the cell at the origin (centre) with no planar translation.

This neighbourhood and the cell at the origin build together a 3·3 square on the original two-dimensional grid. This square will be considered as the largest virtual cell a particle can occupy. The size of this square is therefore 0.9 m·0.9 m, which roughly equals the optimal space which a real pedestrian needs in normal cases. Further, this demands that in these nine cells there should be at most one particle. This can be implemented in that all the eight surrounding cells should be *deactivate*d (written as `d`) for occupation or entrance, once the central cell is occupied. An empty cell, when not deactivated, is considered to be *activate*d (written as `a`).

We wish to point out that a deactivated cell denies occupation or entrance for other particles, this differs from the usual phenomenon of conflict in two senses: first, deactivation or activation depends on other characteristics of the simulation system (e. g. particle density); second, a conflict should be solved by its participants in a bilateral (or multilateral) way, whereas deactivation is unilateral.

We now describe the mechanism of deactivation/activation (abbr.: `d`/`a`) to achieve the effect of variable cell size. Cells outside the Moore neighbourhood (and the origin) with an index 1 will not be affected by this deactivation/activation mechanism, since $0.9\,\text{m} \cdot 0.9\,\text{m}$ is sufficient and optimal for a real pedestrian in a real world scenario. The extreme case $\imath = 0$ refers to an empty two-dimensional index set, this means that no cell needs to be deactivated.

Originally, any neighbourhood index (distance) must be a positive integer. However, an "virtual" index $\imath = 0.5$—since the other four indices $(1, 1)$, $(1, -1)$, $(-1, 1)$ and $(-1, -1)$ will not be counted—would imply a neighbourhood with indices $(0, 1)$, $(0, -1)$, $(1, 0)$ and $(-1, 0)$,[3] this

[1] To conform to the usual definition, we should say that there exists exactly one simulation object in each cell, an empty cell has the state "empty", and in the simulation system, there is an object "`NULL`" in this cell.

[2] Like the majority of the authors, we exclude the index $(0, 0)$ of the origin from the neighbourhood.

[3] Accidentally, this is in fact the von Neumann neighbourhood with index 1 (i. e. the two-dimensional index set of relative positions $\{(0, 1), (0, -1), (1, 0), (-1, 0)\}$). However, other "virtual" Moore neighbourhood indices cannot be expressed in this way.

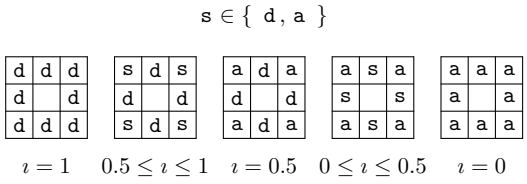

Figure 1: Moore neighbourhood with different indices for the d/a-mechanism. s stands for an unknown state of being either deactivated or activated. The middle-left and the middle-right sub-figures serve as a smooth transition in intervals $\imath \in [0.5, 1]$ and $\imath \in [0, 0.5]$ respectively.

refers to a square with length 0.67 m.[4] This reflects the fact that in a slightly stressed situation, a real pedestrian may wish an exclusive personal space of $0.67\,\mathrm{m} \cdot 0.67\,\mathrm{m}$, and this can be considered as the the middle point between $\imath = 0$ and $\imath = 1$.

The central idea of our ansatz to vary the size of a virtual cell can be split into two steps. First, we extend the possible range of value for the index from $\{0, 1\}$ to the continuous interval $[0, 1]$ of the *virtual* index $\imath$ by constructing a function $\imath$ of the particle density $d_{\mathbf{p},R}$ at position $\mathbf{p} = (r, c)$ with regard to a neighbourhood (or region) R on Ω. This density has $[0, 1]$ as its range and can be estimated for every non-empty cell indexed $\mathbf{p}$ by

$$d_{\mathbf{p},R} = \frac{n_{\blacksquare}(\mathbf{p}, R) - 1}{n_{\square}(\mathbf{p}, R) - 1}, \tag{1}$$

where $n_{\blacksquare}(\mathbf{p}, R)$, $n_{\square}(\mathbf{p}, R)$ denotes the particle and cell numbers[5] in the neighbourhood (or a region) R with regard to position $\mathbf{p}$ respectively. (Obviously, the choice of R should require $n_{\square}(\mathbf{p}, R) > 1$ to avoid division by zero.) From a global perspective (and also to reduce computational complexity), R may also be taken as the whole grid Ω, the trade-off would be a less sensitive system.

In order to be able to decide when $\imath$ takes a value in $(0, 1)$ (i. e. $\imath$ is not an integer, but a real *virtual* index), we choose two threshold values $0 \le d_{\text{low}} < d_{\text{high}} < 1$ for the particle density. There hold $\imath(d_{\text{low}}) = 1$ and $\imath(d_{\text{high}}) = 0$. By the construction of $\imath$ as a simple linear function

$$\imath : [0, 1] \to [0, 1],$$
$$\imath(d_{\mathbf{p},R}) = \begin{cases} 1, & \text{if } 0 \le d_{\mathbf{p},R} \le d_{\text{low}}, \\ \frac{d_{\text{high}} - d_{\mathbf{p},R}}{d_{\text{high}} - d_{\text{low}}}, & \text{if } d_{\text{low}} \le d_{\mathbf{p},R} \le d_{\text{high}}, \\ 0, & \text{if } d_{\text{high}} \le d_{\mathbf{p},R} \le 1, \end{cases} \tag{2}$$

we get the function value of $\imath(d_{\mathbf{p},R})$ as a virtual index $\imath$.

[4] $0.67^2 \doteq \frac{\operatorname{card}\{(0,1),(0,-1),(1,0),(-1,0),(0,0)\}}{\operatorname{card}\{(\Delta r, \Delta c) | \Delta r, \Delta c \in \{-1,0,1\}\}} \cdot 0.9 \cdot 0.9 = \frac{2^2+1}{3^2} \cdot 0.9 \cdot 0.9$.

[5] This is model-dependent. Alternatively, we can integrate the sizes (or masses) of the particles (or generally speaking, the simulation objects).

In the second step, we need to decide whether the cells are to be deactivated or activated, for a given virtual index $\imath$. In other words, we must decide whether s = d or a according to the given $\imath$. Thus $\imath$ serves as a measure for the deactivation or activation. This will be achieved by an appropriate probability distribution. Since a cell must either be deactivated or activated, a Bernoulli distribution with the probability mass function

$$f(\mathtt{s}; p) = \begin{cases} p, & \text{if } \mathtt{s} = \mathtt{d}, \\ 1 - p, & \text{if } \mathtt{s} = \mathtt{a}, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

can be applied here. The parameter p describes the expected probability of a deactivation.

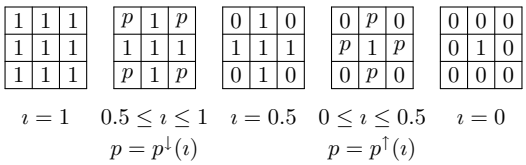

Figure 2: Parameter p of different Bernoulli distributions for different indices for the d/a-mechanism. The cell at the origin is occupied and should therefore always be deactivated. The middle-left and the middle-right sub-figures serve as a smooth transition in intervals $\imath \in [0.5, 1]$ and $\imath \in [0, 0.5]$ respectively.

From Figure 1 we can easily recognize two different transitions from $\imath = 0$ to 0.5 and from $\imath = 0.5$ to 1. We therefore decompose the interval $[0, 1]$ to be $[0, 0.5]$ and $[0.5, 1]$. Obviously, we need two different functions $p^{\uparrow}$ and $p^{\downarrow}$ to control the change of p in (3) in both the two sub-intervals, see Figure 2. p in (3) (cf. Figure 2) can be computed by the functions $p^{\uparrow}$ and $p^{\downarrow}$

$$p := \begin{cases} p^{\uparrow}(i), & \text{if } 0 \le \imath \le 0.5, \\ p^{\downarrow}(i), & \text{if } 0.5 \le \imath \le 1, \end{cases}$$

where $p^{\uparrow} : [0, 0.5] \to [0, 1]$ and $p^{\downarrow} : [0.5, 1] \to [0, 1]$ (cf. Figure 2) are bounded by the conditions

$$p^{\uparrow}(0) = 0, \qquad p^{\uparrow}(0.5) = 1, \tag{4a}$$
$$p^{\downarrow}(1) = 1, \qquad p^{\downarrow}(0.5) = 0. \tag{4b}$$

The transitions in the two sub-intervals request that $p^{\uparrow}$ and $p^{\downarrow}$ be monotonic. Furthermore, we expect that real human pedestrians behave in a rational way, that is, a small change in the environment (e. g. the virtual index $\imath$) only causes small change in the size of the exclusive two-dimensional space they keep for themselves. Any change of this kind should be continuous. We therefore request that $p^{\uparrow}$ and $p^{\downarrow}$ should be continuous, as well.

Examples of appropriate choices for $p^{\uparrow}$ and $p^{\downarrow}$ will be given in the next section. It is obvious that the probability distribution which the cell deactivation/activation obeys in the Moore neighbourhood (of index 1) of an non-empty cell is always well-defined. Since the index $\imath$ in $[0,1]$ reflects the particle density $d_{\mathbf{p},R}$ in the simulation, the magnitude of the repulsion of the individual pedestrians toward others is therefore well-defined as well.

Given a two-dimensional cell index $\mathbf{p} = (r,c)$ on the grid Ω, the parameters of the Bernoulli distributions, which the eight cells in the Moore neighbourhood of index 1 and the cell at the original position (r,c) obey for deactivation/activation, build a matrix which we call *prolongation* matrix of deactivation

$$P^{(r,c)} := \begin{pmatrix} p^{(r,c)}_{-1,-1} & p^{(r,c)}_{-1,0} & p^{(r,c)}_{-1,+1} \\ p^{(r,c)}_{0,-1} & p^{(r,c)}_{0,0} & p^{(r,c)}_{0,+1} \\ p^{(r,c)}_{+1,-1} & p^{(r,c)}_{+1,0} & p^{(r,c)}_{+1,+1} \end{pmatrix}. \tag{5}$$

Obviously, the prolongation matrix $P^{(r,c)}$ depends on the virtual index $\imath$. By the aforesaid construction, it is clear that a prolongation matrix is only well-defined for a non-empty cell positioned at $\mathbf{p} = (r,c)$ on Ω.

Technical note. 1. For $\mathbf{p} = (r,c)$ on the boundary of Ω, some of the components of $P^{(r,c)}$ become meaningless on Ω (when $(r+\Delta r, c+\Delta c) \notin \Omega$ for some $\Delta r, \Delta c \in \{-1,0,1\}$). Later we will see that we only need those components with $(r+\Delta r, c+\Delta c) \in \Omega$ for the calculation.

We note that the components of different prolongation matrices may overlap. Whether to deactivate or activate a non-empty cell at $\mathbf{p} = (r,c)$ is affected by all well-defined components

$$p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c} \tag{6}$$

of up to nine different prolongation matrices $P^{(r-\Delta r, c-\Delta c)}$ for $\Delta r, \Delta c \in \{-1,0,1\}$.

In the sequel, we denote by $(*)$ the condition that the component $p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c}$ is well-defined for a given pair $\Delta r, \Delta c \in \{-1,0,1\}$.

Technical notes. 2. (6) is well-defined if (5) is well-defined, i. e. the cell at $(r-\Delta r, c-\Delta c)$ must not be empty. **3.** If in (6) the relative position $(\Delta r, \Delta c) = (0,0)$ points to a non-empty cell (in other words, the cell at (r,c) is occupied itself), we immediately know that this cell must be deactivated.

3 Implementation

In this section, we discuss aspects of a practical implementation of the model introduced in the preceding section. In particular, the specification of the deactivation probability and the construction of the functions $p^{\uparrow}$ and $p^{\downarrow}$ will be of interest.

3.1 Deactivation probability

Based on the fact of the overlapping prolongation matrices, the cell at $\mathbf{p} = (r,c)$ should be deactivated with a probability of p_d (recall the preceding section) that takes into account the multiple influences of (up to nine) prolongation matrix components $p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c}$ in (6). We call this procedure of obtaining p_d as the probability for deactivation *restriction*.

I. Maximal deactivation

A straightforward strategy consists in maximizing p_d while taking into consideration all possible (up to nine) options, that is,

$$p_d = \begin{cases} 0, & \text{if there exists no combination} \\ & \text{of } \Delta r \text{ and } \Delta c \text{ for which } (*) \\ & \text{is fulfilled,} \\ \max\limits_{(*)\ \text{fulfilled}} p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c}, & \\ & \text{otherwise.} \end{cases} \tag{7}$$

Speaking in a technical way, the deactivation probability p_d will always be overwritten by a larger $p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c}$ for all the combinations of Δr and Δc. We also note that with this implementation, although the size of the simulation system (in the sense of cell numbers) remains unchanged, a large $\imath$ would drastically scale down the actual size (in the sense of the number of the particles) of the simulation system (up to $\frac{1}{9}$), since each particle now deactivates the neighbouring (up to 8) cells, whereas without the `d/a`-mechanism each particle occupies exactly one cell. This effect becomes extremely explicit, when the maximum principle is applied.

On the other hand, (7) guarantees that

$$0 \le p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c} \le 1.$$

This implies that

$$\max_{(*)\ \text{fulfilled}} p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c}$$

is in fact a maximum norm of all the possible $p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c}$s, i. e.

$$\left\| \left(p^{(r-\Delta r, c-\Delta c)}_{\Delta r, \Delta c} \right)_{(*)\ \text{fulfilled}} \right\|_{\infty}.$$

II. Weighted deactivation

We recall the well-known relation

$$\|x\|_\infty \le \|x\|_1 \le n\|x\|_\infty$$

between the 1-norm $\|x\|_1 = |x_1| + \cdots + |x_n|$ and the maximum norm $\|x\|_\infty = \max\{|x_1|, \ldots, |x_n|\}$. This gives rise to a second implementation of the d/a-mechanism.

If we wish to keep the actual size of the simulation system at a similar level as without the d/a-mechanism (cf. Implementation I), we may replace $\|x\|_\infty$ by $\frac{1}{n}\|x\|_1$. Accordingly, we modify the second case of (7) into

$$p_d = \frac{1}{n} \sum_{(*)\ \text{fulfilled}} p_{\Delta r,\Delta c}^{(r-\Delta r,c-\Delta c)}, \tag{8}$$

where n denotes the number of all possible combinations of Δr and Δc,

$$n = \operatorname{card}\{(\Delta r, \Delta c) \mid (*) \text{ is fulfilled}\}.$$

It is easily seen that all the $p_{\Delta r,\Delta c}^{(r-\Delta r,c-\Delta c)}$'s are given the same weight $\frac{1}{n}$.

It is reasonable to give different $p_{\Delta r,\Delta c}^{(r-\Delta r,c-\Delta c)}$ components different weights in computing the deactivation probability p_d. Therefore, we introduce the following *weighting* matrix

$$A = \begin{pmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{pmatrix}$$

with the bounding conditions

$$a_{0,0} > a_{z,0}, a_{0,z} > a_{z,z}, a_{z,-z} > 0, \quad \text{for } z \in \{-1, 1\},$$

and

$$\sum_{r,c\in\{-1,0,1\}} a_{r,c} = 1. \tag{9}$$

Instead of (8), we can now, considering all possible combinations of Δr and Δc, take

$$p_d \;=\; \sum_{(*)\ \text{fulfilled}} a_{\Delta r,\Delta c}\, p_{\Delta r,\Delta c}^{(r-\Delta r,c-\Delta c)}. \tag{10}$$

as a new deactivation probability for an empty cell positioned at (r, c) in the interior of Ω.

Technical notes. 4. It is clear that (10) should be applied on *empty* cells only. Non-empty cells should be considered as deactivated automatically. An empty cell, when not affected by any well-defined (6), should be activated automatically. **5.** Obviously, the construction of the prolongation and restriction matrices and the d/a-mechanism guarantee that p_d from (10) is always in the range $[0, 1]$ for all interior cells of Ω.

Now we consider the choice of the matrix components of A (i. e. the weights). We have mentioned earlier that p_d is affected by components of different prolongation matrices, and every of these components has a different matrix index (i. e. position) in the corresponding prolongation matrix. The distance of this position to the origin $(0, 0)$ can thus be used to decide the weight by which p_d undergoes the influence of this prolongation matrix component.

We note that for $z_1, z_2 \in \{-1, 1\}$, we have

$$\|(z_1, z_1) - (0, 0)\|_2 = \|(z_2, -z_2) - (0, 0)\|_2 = \sqrt{2},$$
$$\|(z_1, 0) - (0, 0)\|_2 = \|(0, \quad z_2) - (0, 0)\|_2 = 1.$$

The interpretation of these numbers as distances of two-dimensional indices representing relative cell distances in a $3 \cdot 3$ square leads us to an appropriate choice for the components of A. It is reasonable to request that A has the form

$$A = \begin{pmatrix} a_4 & a_2 & a_4 \\ a_2 & a_1 & a_2 \\ a_4 & a_2 & a_4 \end{pmatrix}$$

with

$$\frac{a_4}{a_2} = \frac{1}{(\sqrt{2})^2} = \frac{1}{2}; \tag{11}$$

and additionally, we can request that

$$\frac{a_2}{a_1} = \frac{1}{(1+1)^2} = \frac{1}{4}, \tag{12}$$

since the distance between $(z, 0)$ or $(0, z)$ (for $z \in \{-1, 1\}$) and $(0, 0)$ is the unit length. The square function applied in (11) and (12) originates from Newtonian physics of force and distance.

Hence, we have $a_1 : a_2 : a_4 = 1 : \frac{1}{4} : \frac{1}{8}$, which declares the interrelationship among the weights of different prolongation matrix components in computing p_d. Rescaling these numbers in order to satisfy (9) leads to

$$a_{0,0} = \tfrac{2}{5} \qquad\qquad = a_1, \tag{13a}$$
$$a_{1,0} = a_{-1,0} = a_{0,1} \quad = a_{0,-1} \quad = \tfrac{1}{10} = a_2, \tag{13b}$$
$$a_{1,1} = a_{-1,1} = a_{1,-1} = a_{-1,-1} = \tfrac{1}{20} = a_4. \tag{13c}$$

We call A *symmetric*, if $a_{1,0} = a_{-1,0} = a_{0,1} = a_{0,-1}$ and $a_{1,1} = a_{-1,1} = a_{1,-1} = a_{-1,-1}$.

Technical notes. 6. The boundary of Ω plays a less significant role, nevertheless we would like to give a generalized form of (10) for an empty cell positioned at (r, c) on Ω:

$$p_d = s_{r,c} \sum_{(*)\ \text{fulfilled}} a_{\Delta r,\Delta c}\, p_{\Delta r,\Delta c}^{(r-\Delta r,c-\Delta c)}, \tag{10'}$$

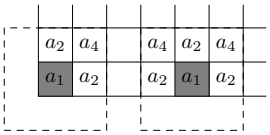

Figure 3: Two trivial cases of the rescaling factor $s_{r,c}$ on the boundary of a regular grid Ω of rectangular shape.

where $s_{r,c}$ is a rescaling factor taking into consideration the weights of different components in A. For all interior cells there is $s_{r,c} = 1$. Apparently, $s_{r,c}$ is not independent of the geometric form (shape) of Ω. For example, in case of an Ω of rectangular shape in combination with a symmetric restriction matrix A given by (13a), (13b) and (13c), the rescaling factor $s_{r,c}$ for the four vertex cells of Ω can be set to $\frac{1}{a_1+2a_2+a_4}$, and for the non-vertex cells on the four edges of Ω, to $\frac{1}{a_1+3a_2+2a_4}$, respectively (cf. Figure 3), and this may be the only reasonable choice, if no additional information concerning the geometry is provided. **7.** Because of the construction of the rescaling factor $s_{r,c}$ in (10') (cf. technical note 5) the value p_d of a deactivation probability computed by (10') might theoretically be larger than 1. In this case, p_d must be reset to 1.

3.2 Construction of $p^{\uparrow}$ and $p^{\downarrow}$

Finally, we discuss the construction of $p^{\uparrow}$ and $p^{\downarrow}$. (4a) and (4b) describe the situations for $\imath = 1$, $\imath = 0.5$ and $\imath = 0$ (cf. Figure 2). To ensure a continuous transition for $0 \le \imath \le 1$, we can construct the monotonic $p^{\uparrow}$ and $p^{\downarrow}$ as a linear function

$$p^{\uparrow}(\imath) = \quad 2\imath, \tag{14a}$$
$$p^{\downarrow}(\imath) = \quad 2\imath - 1; \tag{14b}$$

or a quadratic one,

$$p^{\uparrow}(\imath) = -4\imath^2 + 4\imath, \tag{15a}$$
$$p^{\downarrow}(\imath) = \quad 4\imath^2 - 4\imath + 1, \tag{15b}$$

or a function of even higher order. Although both variants are constructed by continuous functions, in comparison to (14a) and (14b), (15a) and (15b) due to their quadratic form render a smoother transition at $\imath = 0.5$. In the simulation tests, they do not manifest significant distinctions.[6]

[6]The reason is that for the "s" items (cf. Figure 2) the relative difference of (15a) to (14a) (and (15b) to (14b)) is small, this can be estimated by

$$\frac{\int_0^{\frac{1}{2}} (-4x^2 + 4x - 2x)\,\mathrm{d}x}{\int_0^{\frac{1}{2}} 2x\,\mathrm{d}x} = \frac{1}{3}.$$

In Implementation II, this difference undergoes a further weighting of a_2.

For our tests, (14a) and (14b) of the simpler variant have been applied.

3.3 Algorithmic concept

So far, for every single cell we know that either it should be activated unconditionally or we know the probability to deactivate it. Therefore, the simulation can be considered as a pipelining cycle, as shown in Figure 4.

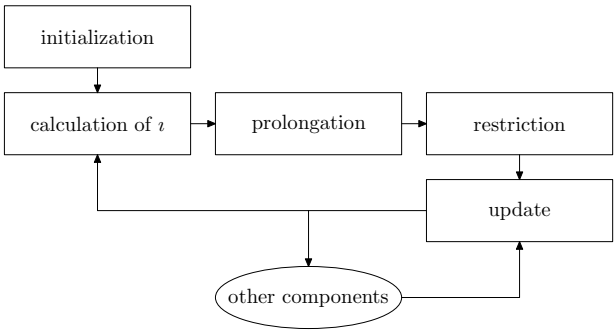

Figure 4: Simulation cycle as a pipeline.

We wish to point out that our model as a cycle of "calculation of $\imath$-prolongation-restriction-update" is model-independent. It can be combined with any other model provided there is a compatible grid defined for the geometry where the virtual index $\imath$ can be constructed and the prolongation/restriction is well-defined.

We have mentioned previously that deactivation/activation differs from conflict management. In Figure 4 the latter is carried out as part of the model-dependent module "other components". From a technical perspective, we first decide the deactivation/activation of the cells and then check the availability of the destination cell to avoid all possible conflicts. However, a possible conflict implies that there exists one non-empty cell (or more) in the neighbourhood, and this will add to the possibility of a deactivation p_d for the destination cell where there would be a possible conflict, in some cases, the conflict management becomes merely unnecessary.

4 Experimental Results

First, we give some examples of the two implementations for the calculation of the deactivation probability p_d of the last section.

Different combinations of d_{high} and d_{low} have different impacts on the simulation system. It is worthwhile to point out that a configuration

$$d_{\text{low}} \to 0,$$
$$d_{\text{high}} \to d_{\text{low}},$$

gives a good approximation for a system dismounted of the d/a-mechanism, since the computed $\imath$ tends to be 0 constantly.

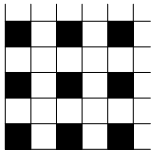

Figure 5: Ideal case (drawn partially) in which every particle (coloured black) takes an exclusive space for $3 \cdot 3$ grid cells.

Additionally, the choice of R in (1) plays a role, since $\imath$ depends on $d_{\mathbf{p},R}$. From Figure 5 we clearly see that different counting methods deliver surprisingly different particle densities: the global[7] density is $\frac{1}{4}$; calculated columnwise of a non-empty column, it is $\frac{1}{2}$; calculated cellwise of a non-empty cell, it becomes $\frac{1}{9}$ in the Moore neighbourhood with index 1.

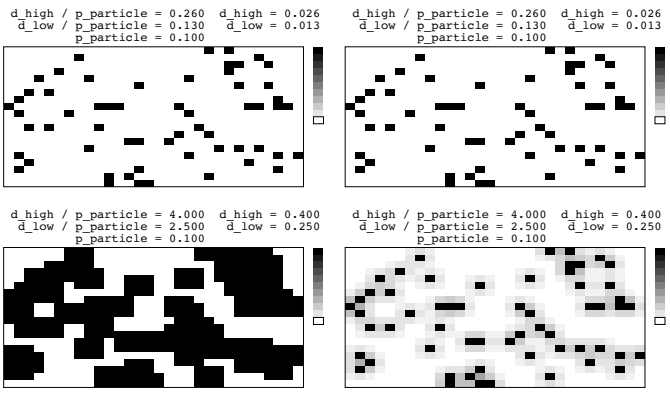

Figure 6: Some bad configurations. Top: a too small value of d_{low} approximates the situation in which the d/a-mechanism is not applied. Bottom: a too large value of d_{low} induces excessively large deactivation in the neighbourhood.

We test our two implementations on an "empty" rectangular grid of $20 \cdot 30$ cells without any interior obstacles, entrances or exits. The tests are solely meant to be static. The particles are produced with a probability of $p_{\text{particle}} = 0.1$. Since the particles are produced on a random basis, there will be no substantial difference among the various choices of R in computing $d_{\mathbf{p},R}$ (cf. (1)) (provided the grid is sufficiently large). Therefore, we choose to calculate the particle density columnwise. Obviously, the probability p_{particle} with which the particles are produced can be seen as an estimate of overall particle density. Some examples of different configurations are given in Figures 6 and 7, where non-empty and empty grid cells are drawn in black and white respectively. In addition, probabilities for an activation are drawn in a linear grayscale; accordingly, black stands for a deactivation and white for an activation. Implementation I and II are presented in the left and right sub-figures respectively. As mentioned earlier, due to the high deactivation probabilities for the grid cells, Implementation I considerably reduces the actual size of the system.

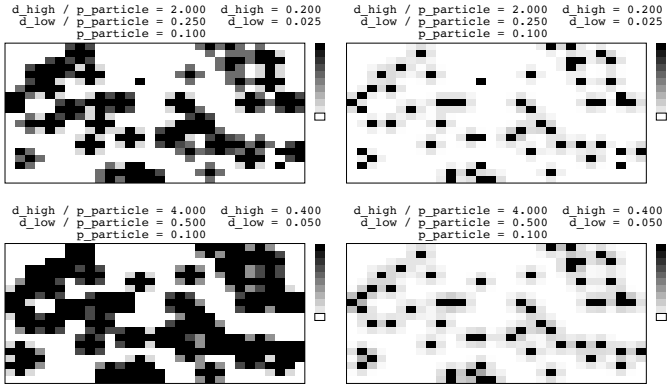

Figure 7: Configurations of moderate d_{high} and d_{low} values.

In comparison to Figure 6, Figure 7 offers two acceptable configurations of d_{high} and d_{low}, from which we see that a moderate small d_{low} (smaller than p_{particle}) and a moderate large d_{high} (multiple of p_{particle}, but less than 1) would be a good combination, because the cases $\imath = 0$ and $\imath = 1$ are given consideration, and at the same time the "spectrum" of $\imath \in [0, 1]$ is sufficiently visible as well.

[7] The grid is considered to be infinite, this is assumed for the columnwise density, too.

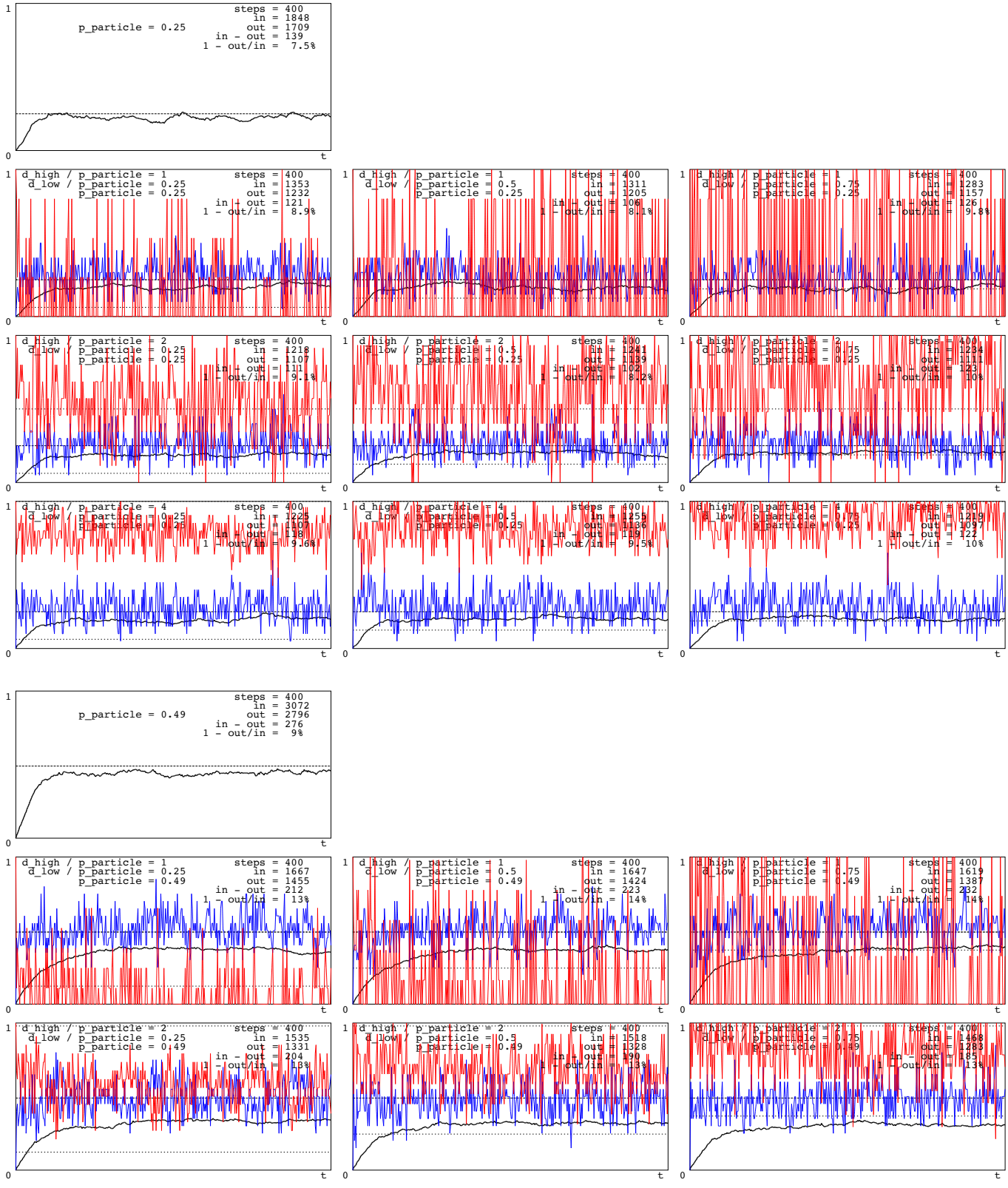

Figure 8: Simple tests without and with the d/a-mechanism (Implementation II only, in combination with different d_{high} and d_{low} configurations). Columnwise density is applied. Particle densities and i values of the first column (left border of Ω) are drawn in blue and red respectively. The global (i. e. $R = \Omega$ in (1)) particle densities, in a time sequence, are drawn as curves. Particle production probabilities p_{particle} (and d_{high}, d_{low}, as with the d/a-mechanism) are drawn as dashed lines.

Now we apply our model on a very simple test case. Again, we choose the grid Ω to be of rectangular shape: $\Omega = \{1, \ldots, n_{\text{row}}\} \times \{1, \ldots, n_{\text{col}}\}$ with $n_{\text{row}} = 20$, $n_{\text{col}} = 30$ denoting the row and column numbers respectively. On the left boundary of the grid, i. e. $\{(r, 1) \mid r = 1, \ldots, n_{\text{row}}\}$, particles will be produced on a random basis (obeying a predefined Bernoulli distribution) in each simulation step ($p_{\text{particle}} = 0.25$ and 0.49). We do not allow further simulation objects.

On a second random basis, the particles are allowed to take a step choice of going one step forward, two steps forward, one step right, one step left or no movement at all. In the simulation, theses probabilities are set to be 0.33, 0.42, 0.15, 0.075 and 0.025.

In case of a conflict, maximal step choice is applied: the particles undertake one step forward, one step right, one step left, when they are assigned such a random step choice and the destination cell is not blocked, otherwise, stay with no movement; in case of a step choice of two cells forward, the particles takes the largest possible movement. The particles leave the grid Ω on arriving its right border. Except this, the particles are not allowed to go out of the grid boundary. The rate of particle flow in the system is of crucial interest.

Particles are produced randomly on the left border of Ω and flow out of the system on the right border. The global particle densities (in a time sequence of 400 simulation steps) have been calculated under different parameter conditions. Since the change in the first column exerts an overall influence on the whole system, particle densities and $\imath$ values in the first column are shown additionally in Figure 8.

Since the particles flow into the simulation system on the left border (the first column) of Ω, columnwise density calculation is reasonable. From Figure 8 we also notice that besides the global particle density, the flow-in and flow-out numbers of the particles differ from each other. This is because in the first column where particles are randomly produced (in other words, flow in), activation and deactivation must be considered. In a nontrivial sense (on Ω), activation and deactivation have an explicit influence of the general moving velocity of the particles.

In the test cases of Figure 8 of two different p_{particle} values, there are 139 and 276 particles left in the system without the d/a-mechanism, whereas with it, the numbers of the remaining particles (derived by subtracting total flow-out number from total flow-in number) vary from 102 to 126 and from 185 to 232 respectively. The actual size of the simulation system remains roughly the same. However, with a higher $\imath$ value (drawn in red in the figure), cell deactivation tends to take place more frequently. This results in an unavoidably retarded flow-in.

Figure 9 illustrates the difference between a simple con-

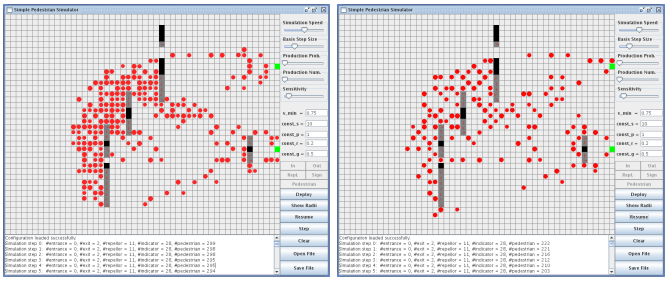

Figure 9: Two screenshots. Left: An example produced by the program from [3]. Right: A similar example with the d/a-mechanism.

flict management and the additional application of the above d/a-mechanism. The example on the left is constructed by the program from [3]. Green and black cells represent exits and interior obstacles (walls) respectively. Particles, flowing toward own destinations, are shown as red points. In [3], the grid cell was chosen to be $0.5\,\text{m} \cdot 0.5\,\text{m}$. In the simulation, once this value is set, it can no longer be modified. Without the d/a-mechanism it would impossible to describe the situation where the real pedestrians request an exclusive minimal space of different size. The right sub-figure assumes a statistically larger (but variable) exclusive minimal space for the pedestrians. In this case, the reader recall that the actual size of the simulation system is reduced by the application of the d/a-mechanism, since the physical size of the grid cells remains unchanged, and at the same time, the particles take a larger space for themselves. To be able to describe the change of the exclusive minimal space for the pedestrians, we only need to construct $\imath$ in a reasonable way (cf. next section).

5 Further Discussion

Our d/a-mechanism is defined in the Moore neighbourhood with index 1 (cf. Figure 1). Generally speaking, any arbitrary index set can be used, if the corresponding prolongation and restriction procedures are well-defined. However, considering the human sociological aspect, the choice of Moore neighbourhood with index 1 is adequate and reasonable, although the change of the activation probabilities (e. g. in Figures 6 and 7) might not appear to be continuous (from cell to cell, in a vicinity of a non-empty cell larger than the aforesaid Moore neighbourhood of index 1). A further research topic would be, when provided with sufficient empirical data, the construction of $\imath$ (with the help of d_{high} and d_{low}) as a dynamical function of the parameters, depending on further system information besides the particle density in the simulation.

We may notice the fact that real human pedestrians are

much more (when they are motivated toward some certain destination, that is, they possess a temporal moving direction) affected by the environmental settings in front of them than on the sides and behind them. In [3], an anisotropic variant of neighbourhood in elliptical shape has been suggested, so in our model, unsymmetric prolongation matrix deserves future investigation, with the matrix components (in terms of the its difference as a two dimensional index to the origin) closer (or aligning) to the current particle moving direction being given more weight. This unsymmetric prolongation matrix should be nontrivial, if we take a look of a waiting queue in the real world, in certain directions (i. e. on the sides) cell deactivation shall be undertaken whereas the space is physically free, because people (in a well-formed context) choose to align with the queue direction.

Acknowledgement. The authors gratefully acknowledge support of the present work by the regional government of Berlin within the grant program *ProFIT* (project 10134782) partially financed by the *European Fund for Regional Development (EFRE).*

References

[1] Bärwolff, G., Chen, M.-J., Schwandt, H. and Slawig, T., *Simulation of pedestrian flow for traffic control systems*, in Lin, E. Y. (ed.), Proceedings of the sixth International Conference on Information and Management Sciences, Lhasa, China, 1.–6. July 2007, California Polytechnic State University, Series on Information and Management Sciences, Volume 6 (2007), pp. 708–714

[2] Burstedde, C., Klauck, K., Schadschneider, A. and Zittartz, J., *Simulation of pedestrian dynamics using a two-dimensional cellular automaton*, Physica A 295 (2001), pp. 507–525

[3] Chen, M.-J., Bärwolff, G. and Schwandt, H., *A derived grid-based model for simulation of pedestrian flow*, submitted (2008)

[4] Esser, J. and Schreckenberg, M., *Microscopic simulation of urban traffic based on cellular automata*, Int. J. of Mod. Phys. C. 8 (5) (1997), pp. 1025–1036

[5] Hamacher, H. W. and Tjandra, S. A., *Mathematical modelling of evacuation problems: a state of the art*, in [17], pp. 227–266

[6] Helbing, D., Verkehrsdynamik: Neue Physikalische Modellierungskonzepte, SpringerVerlag, Berlin (1997)

[7] Helbing, D., Farkas, I. and Vicsek, T., *Simulating dynamical features of escape panic*, Nature 407 (2000), pp. 487–490

[8] Helbing, D., Herrmann, H. J., Schreckenberg, M. and Wolf, D. E. (eds.), Traffic and Granular Flow '99, Springer-Verlag, Berlin (2000)

[9] Keßel, A., Klüpfel, H., Wahle, J. and Schreckenberg, M., *Microscopic simulation of pedestrian crowd motion*, in [17], pp. 193–200

[10] Kirchner, A., Nishinari, K. and Schadschneider, A., *Friction effects and clogging in a cellular automaton model for pedestrian dynamics*, Phys. Rev. E 67 (5), 056122 (2003)

[11] Klüpfel, H. L., *A cellular automaton model for crowd movement and egress simulation*, dissertation (2003)

[12] Nagel, K. and Schreckenberg, M., *A cellular automaton model for freeway traffic*, J. Phys. I France 2 (1992), pp. 2221–2229

[13] Nishinari, K., Kirchner, A., Namazi, A. and Schadschneider, A., *Extended floor field CA model for evacuation dynamics*, IEICE Trans. Inf. & Syst. E87-D (3) (2004), pp. 726–732

[14] Schadschneider, A., *Cellular automaton approach to pedestrian dynamics - theory*, in [17], pp. 75–85

[15] Schadschneider, A. and Schreckenberg, M., *Cellular automaton models and traffic flow*, J. Phys. A. Math. Gen. 26 (15) (1993), pp. L679–L683

[16] Schreckenberg, M. and Wolf, D. E. (eds.), Traffic and Granular Flow '97, Springer-Verlag, Singapore (1998)

[17] Schreckenberg, M. and Sharma, S. D. (eds.), Pedestrian and Evacuation Dynamics, Springer-Verlag, Berlin (2002)

[18] Zhang, J., Wang, H. and Li, P., *Cellular automata modeling of pedestrian's crossing dynamics*, J. Zhejiang Univ. Sci. 5 (7), Hangzhou, China (2004), pp. 835–840

Minjie Chen received the degree of Dipl.-Inf. from the University of Leipzig in 2006. He works in the field of applied informatics. Since 2007 he is working on a project for mathematical and software solutions of traffic control.

Günter Bärwolff holds the position of apl. Prof. of Mathematics at the Technical University of Berlin since 2002. He is working in the field of numerical mathematics, optimization with PDE's, parallel algorithms, large systems, computational fluid dynamics and is responsible for several projects for mathematical and software solutions for applications in the area of traffic control.

Hartmut Schwandt holds the position of apl. Prof. of Mathematics at the Technical University of Berlin since 1998. He is working in the field of numerical mathematics, parallel algorithms, large systems and interval mathematics and is responsible for several projects for mathematical and software solutions for applications in the area of traffic control.