Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

A derived grid-based model for simulation of pedestrian flow*

Minjie CHEN, Günter BÄRWOLFF, Hartmut SCHWANDT
(*Institut für Mathematik, Technische Universität Berlin, Berlin 10623, Germany*)
E-mail: {minjie.chen, baerwolf, schwandt}@math.tu-berlin.de

Received Jan. 17, 2008; Revision accepted Sept. 22, 2008; Crosschecked Dec. 26, 2008

Abstract: We present a derived grid-based model for the simulation of pedestrian flow. Interactions among pedestrians are considered as the result of forces within a certain neighbourhood. Unlike the social force model, the forces here, as in Newtonian physics, are proportional to the inverse of the square of the distance. Despite the notion of neighbourhood and the underlying grid, this model differs from the existing cellular automaton (CA) models in that the pedestrians are treated as individuals. Bresenham's algorithm for line rastering is applied in the step calculation.

Key words: Bresenham's algorithm, Cellular automaton (CA), Pedestrian dynamics, Social force model
doi:10.1631/jzus.A0820049 **Document code:** A **CLC number:** TP391

INTRODUCTION

Computer simulation of pedestrian and traffic flow has been intensively investigated (Helbing, 1997; Schreckenberg and Wolf, 1998; Helbing *et al*., 2000b; Schreckenberg and Sharma, 2002; Hoogendoorn *et al*., 2005; Schadschneider *et al*., 2007; Waldau *et al*., 2007) since the last decade. The modeling is either macroscopic, emphasizing the global properties of the pedestrians, or microscopic, with the pedestrians treated as individuals. From a macroscopic perspective, the classical Maxwell-Boltzmann theory of molecular systems serves well to describe the distribution of pedestrian movement (Palmer and Bailey, 1975). As microscopic models, cellular automata (CA) are often encountered.

In the pioneering work (Nagel and Schreckenberg, 1992; Schadschneider and Schreckenberg, 1993), the Nagel-Schreckenberg model of one-dimensional (1D) CA was proposed to simulate traffic flow in a closed (circle-shaped) system. This model was able to capture some essential characteristics of one-lane traffic systems and was later extended by Esser and Schreckenberg (1997) to include multi-lane systems. Some researchers introduced two-dimensional (2D) CA models for the simulation of pedestrian flow (Burstedde *et al*., 2001; Keßel *et al*., 2002; Klüpfel, 2003). These models dealt with the case $v_{\max}=1$ (common simplified form for $v_{\max}\cdot\Delta t=1$ cell size in one dimension, where Δt denotes the step time span of the simulation clock); i.e., in one simulation step, the pedestrians are allowed one-cell-moves exclusively: the possible moves that a pedestrian at the position $\boldsymbol{p}=(p_x, p_y)$ may take are restricted to being toward the immediate neighbours at $(p_x+\Delta x, p_y+\Delta y)$ with the choice $\Delta x, \Delta y\in\{-1, 0, 1\}$, $(\Delta x, \Delta y)\neq(0, 0)$. In the sequel, pedestrians will be called 'particles'.

A straightforward example for $v_{\max}=1$ m/s was (Keßel *et al*., 2002), in which the particles took exactly one of the three step choices of going forward ($\rightarrow$), upward ($\uparrow$), or downward ($\downarrow$) on a 2D grid. The probabilities for these choices were constructed according to the information collected in the neighbouring cells with the relative position $(\Delta x, \Delta y)\in\{(1, 0), (-1, 0), (0, 1), (0, -1)\}$, and a so-called 'drifting variable' $D\in[0, 1]$:

* Project (No. 10134782) supported by the Regional Government of Berlin within the Grant Program ProFIT partially financed by the European Fund for Regional Development (EFRE)

$$P(\rightarrow) = D + \frac{1-D}{U},$$
$$P(\uparrow) = P(\downarrow) = \frac{1-D}{U},$$

where U is the number of unoccupied neighbouring cells. The drifting variable D was assumed to be an empirical control variable under different geometric circumstances, so that the relative magnitudes of the probabilities in the three walking directions can be decided in accordance with the geometry in advance.

This demands implicitly a normalization in accordance with $\sum P$ (Table 1). We observe that a back-step is not allowed in this model. Additionally, the probability that a particle takes a stop as a step choice is implied to be 0; a particle takes a real stop in the simulation when denied of any movement as the result of a conflict solution in the parallel update.

Table 1 Probabilities for the step choices according to (Keßel *et al*., 2002)

U	$P(\rightarrow)$	$P(\uparrow)$, $P(\downarrow)$	$\sum P$
1	1	$1-D$	$3-2D$
2	$1/2+D/2$	$1/2-D/2$	$3/2-D/2$
3	$1/3+2D/3$	$1/3-D/3$	1
4	$1/4+3D/4$	$1/4-D/4$	$3/4+D/4$

In the CA model for fluid dynamics and particle transport (Chopard and Masselot, 1999), once a particle's ideal movement is known, four numbers $p^{+1,0}$, $p^{0,+1}$, $p^{-1,0}$, $p^{0,-1} \in [0, 1]$ (the superscripts +1, −1 and 0 stand for one positive, one negative and zero unit on the x-, y-axis respectively) can be computed with the information of local particle velocity to choose a neighboring cell as a step choice for it. For instance, when a particle should have an ideal move from position $\boldsymbol{p}$ to $\boldsymbol{p}+\boldsymbol{w}$ with $\boldsymbol{w}=(w_x, w_y)$ for $w_x>0$, $w_y>0$ (in this case, only $p^{+1,0}$ and $p^{0,+1}$ will be needed, and the other two are by original definition 0), the four numbers $p^{+1,0}(1-p^{0,+1})$, $p^{+1,0}p^{0,+1}$, $p^{0,+1}(1-p^{+1,0})$, and $(1-p^{+1,0})(1-p^{0,+1})$ could be taken as the probabilities that a particle moves one cell in x direction, one cell in both x and y directions, one cell in y direction, or stops. An obvious advantage is that these four numbers always add up to 1. Thus, this can be interpreted as an approximation on the lattice board, when the new position $\boldsymbol{p}+\boldsymbol{w}$ does not correspond to a natural lattice node, which is normally the case.

Also in the CA models, the ansatz of floor fields (static and dynamic) by (Burstedde *et al*., 2001; Schadschneider, 2002; Nishinari *et al*., 2004) took into consideration the geometry of the simulation (static and time-independent) and the dynamic interactions among the particles. A combination of these two fields renders a step choice among the eight immediate neighbours.

In the present paper we wish to study the general case $v_{\max} \geq 1$. This makes possible the downsizing of the grid cells, and consequently the particles may be considered as more than just sharing the same physical characteristics (e.g., velocity). Meanwhile, cell information in a chosen area (the generalized case for the surrounding cells, called 'neighbourhood' later) will be considered.

MODEL

In our model, the geometry is defined on a 2D Cartesian grid $\Omega=[0, l_x n_x]\times[0, l_y n_y] \subset \mathbb{R}^2$, with l_x, l_y and n_x, n_y being the length, width and the numbers of the grid cells on the x-, y-axis respectively. We leave out the unit of measurement for length here to simplify the notation. In some cases, the set Ω may be re-scaled by l_x and l_y and written in the discrete form $\Omega=\{1, 2, \ldots, n_x\}\times\{1, 2, \ldots, n_y\}$, as a set of grid cells. In a broader sense, the grid is not confined to be of rectangular shape.

Palmer and Bailey (1975) argued that the pedestrian velocities (in an undisturbed situation) can be considered as a normal distribution; furthermore, the average velocities of male and female pedestrians in the age group from thirty to fifty were considered to fit into the linear regressions $(1.5637-0.806x)$ m/s and $(1.4334-0.806x)$ m/s respectively, with a linear scale x between −1 (for age 30) and 1 (for age 50). An estimate of the average velocity for all the particles would be

$$\bar{v} = 1.5 \text{ m/s} \qquad (1)$$

accordingly. In the current paper, relative velocity (measured by $\bar{v}$) will be used as well.

In most of the CA models (Burstedde *et al*., 2001; Keßel *et al*., 2002), the cell size was set to be 0.4 m× 0.4 m, with a time span Δt=0.3 s (Burstedde *et al*., 2001). This would correspond to an average velocity

of 1.33 m/s of the pedestrians. Predtetschenski and Milinski (1971) provided us with a detailed list of demanded spaces for people with different physical conditions. It was also claimed that a minimum width of 0.6 m for undisturbed movement was necessary. As a compromise, we choose l_x=l_y=0.5 m for our model, which roughly reflects the minimum space a real pedestrian claims in normal cases. We also choose a somehow larger time span, Δt=1 s [The webpage http://www.humanbenchmark.com/test/reactiontime/index.php (accessed September 22, 2008) provides us with an interesting human reaction test. According to this test, the average person needs 0.2~0.3 s reaction time for a color signal change. We assume that the reaction accounts for roughly one fourth of the time span of a simulation step.], because it is unlikely that a time span as short as 0.3 s would be enough for a real pedestrian to gather information, make his/her own decision and execute this decision. In combination with the mean velocity observed by Palmer and Bailey (1975), the time span of our choice implies a step length of 1.5/0.5×1=3 cells. This gives rise to the investigation of $v_{\max}$>1 m/s on the grid.

Naturally, we should request that the particles not go beyond the boundary of Ω. We are not interested in the states of all the grid cells but the change of the states of certain grid cells by which the flow of the particles is represented. We do not give universal transition rules for all the cells; unlike the common CA models, the dynamics of the particles are considered as the result of their interactions. Additionally, we have the following basic assumptions:

(1) The contents of the cells will be called 'objects'. An empty cell is of the object type 'empty'.

(2) The particles may differ from each other. Their individual characteristics can be arbitrarily pre-configured. In general, this applies to all non-empty objects.

(3) The particles possess no *a priori* knowledge of reaching a local optimum. A single particle with a fixed destination collects the environmental information at each simulation step to make the step choice.

(4) The particles are neither 'cooperative' (to reach an overall optimum) nor 'competitive' (so that the overall optimization will be hindered by local optima).

Object and neighbourhood

A non-empty grid cell can be of one of the following object types:

(1) Entrance: $\boldsymbol{i}_k \in \Omega$, k=1, 2, …, n_{in}; we write $I = \bigcup_k \boldsymbol{i}_k$ as the set of all entrances.

(2) Exit (or destination): $\boldsymbol{o}_k \in \Omega$, k=1, 2, …, n_{out}; we write $O = \bigcup_k \boldsymbol{o}_k$ as the set of all exits.

(3) Repellor (or wall): $\boldsymbol{r}_k \in \Omega$, k=1, 2, …, n_{rep}, which blocks the pedestrian movement; we write $R = \bigcup_k \boldsymbol{r}_k$ as the set of all repellors.

(4) Indicator (or sign board): s_k=($\boldsymbol{s}_k$, α_k), $\boldsymbol{s}_k \in \Omega$, $\alpha_k \in [0, 2\pi)$, k=1, 2, …, n_{ind}; α_k as an angle gives a static influence on the particles for the ideal flow direction, which is the 'indicating' effect that s_k has on the particles.

Examples of repellor and indicator can be found later in Fig.3. As a fixed object, an indicator can be considered as a special repellor as well; however, we assume that the 'indicating' effect (expressed by α_k) is overwhelming in this context.

(5) Particle: p_k, k=1, 2, …, n_{ped}, the major role of the simulation. Besides its position $\boldsymbol{p}$ on the grid Ω, a particle p is associated with an entrance $\boldsymbol{i}_p \in I$, a destination $\boldsymbol{o}_k \in O$, an individual mass m_p and a relative velocity r_p (measured in the estimated average value $\bar{v}$, cf. Eq.(1)),

$$p=(m_p, \boldsymbol{p}, \boldsymbol{o}_p, r_p). \tag{2}$$

The mass m_p and velocity r_p can be set constant or configured to obey a certain probability distribution. Obviously, the mathematical expectation of the relative velocity $\sum r_p/n_{\text{ped}}$ should be 1 m/s. Furthermore, we assume that the maximum velocity attainable by the particles is 300% (for this we write q=3 as a constant) of this relative velocity r_p. This implies that, if we set $r_p \in [0.5, 1.5]$, the fastest particles move at an average velocity of 2.25 m/s and the maximum velocity, observable with these fastest particles, would reach 6.75 m/s (please compare the world record for the 100-m sprint, which is approximately 10 s). We introduce the superscript '(n)' to denote the nth time step. We may omit this superscript, when it is unambiguous in the context. Obviously, the start position $\boldsymbol{p}^{(0)}$ depends on $\boldsymbol{i}_p$,

$$\boldsymbol{p}^{(0)}=\boldsymbol{i}_p.$$

We write $P = \bigcup_k p_k$ as the set of all particles.

The interior wall can be considered as a repellor. We assume that repellors are global, which means they have the same unconditional blocking effects on the particles.

We recognize the fact that a remote object exerts weak or merely neglectable influences on the others. Therefore, we consider the index set of relative positions within a certain distance, also called 'neighbourhood'. The Moore neighbourhood with a distance value (also called 'radius') is defined as

$$M_r(x_0, y_0)=\{(x, y)|x-x_0, y-y_0\in\{-r, \ldots, 0, \ldots, r\}, (x, y)\neq(x_0, y_0)\}, \quad (3)$$

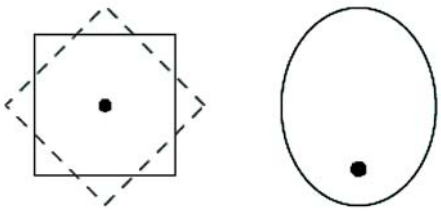
and the von Neumann neighbourhood as

$$N_r(x_0, y_0)=\{(x, y)||x-x_0|+|y-y_0|\in\{1, 2, \ldots, r\}, |x-x_0|, |y-y_0|\in\mathbb{N}\}. \quad (4)$$

In some texts, the index tuple (x_0, y_0) of the original position is not excluded from the neighbourhood. The same applies to the von Neumann neighbourhood. However, this makes no practical difference in our model.

Since real human pedestrians care more about the environmental settings in front of them than on the sides and behind them, an elliptical-shaped variant of the neighbourhood may be another choice, in which the original position (x_0, y_0) overlaps one of the foci and the particle's local moving direction lies on the major axis of this ellipse.

Fig.1 shows three simple examples of the neighbourhood. We would use the traditional variant Eq.(4) in the implementation.



(a) (b)

Fig.1 The idea of neighbourhood

(a) Moore neighbourhood (solid line) and von Neumann neighbourhood (dashed line); (b) Neighbourhood in an elliptical shape

Interaction and force

The well-established social force model (Helbing *et al*., 2000a) assumed that the behaviour of the individual pedestrians was induced by the impacts (called 'social force') from the other pedestrians or objects. The social force $\boldsymbol{f}_{ij}$, in its simplest form, of two particles i and j not touching each other, is given as

$$\boldsymbol{f}_{ij} = A_i \mathrm{e}^{(r_{ij}-d_{ij})/B_i}\boldsymbol{n}_{ij} \quad (5)$$

with positive constants A_i and B_i, where d_{ij} denotes the distance between particles i and j, r_{ij} the sum of the radii of particles i and j ($r_{ij}<d_{ij}$, since the two particles do not touch each other), and $\boldsymbol{n}_{ij}$ the normalized vector pointing from the position of particle j to that of particle i. Eq.(5) shows a 'fast decay', i.e., the impact from a remote object decreases exponentially; at the same time, the integral

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\mathrm{e}^{-\sqrt{x^2+y^2}}\,\mathrm{d}x\mathrm{d}y$$

exists (i.e., 2π), which, after some parameter transformation, can be understood as the magnitude of the overall force on a particle in an unbounded region with an unbounded particle density. However, since in the current model all the interactions are considered within a certain neighbourhood, we do not request the existence of the aforesaid improper integral. We prefer, therefore, a form derived from Newtonian physics

$$\frac{Cm_im_j}{d_{ij}^2}\boldsymbol{n}_{ij} \quad (6)$$

with a positive constant C, where m_i and m_j are the masses of the particles.

In Eq.(6), the magnitude of the force is proportional to the inverse of the square of the distance. However, this fundamental difference does not prevent Eqs.(5) and (6) from exhibiting sufficient local similarity (for a well-chosen C respecting A_i, B_i and r_{ij}), see Fig.2. Eq.(6) is the abstract form of the interactions among the simulation objects in our model. In the sequel, the distance between two simulation objects will be written as the Euclidean norm $\|\cdot\|_2$ (shortened as $\|\cdot\|$) of the vectorial difference of their positions.

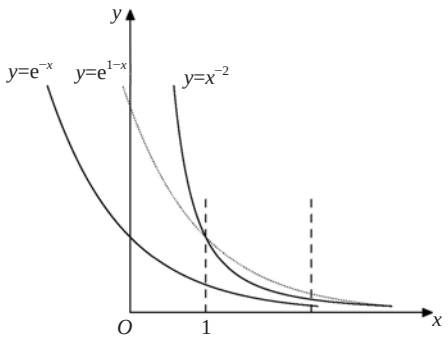


Fig.2 The function $y=e^{-x}$, its translation $y=e^{1-x}$ (a special case of Eq.(5) with $A_i=1$, $r_{ij}=B_i=1$, and $x=d_{ij}$), and the function $y=x^{-2}$
The interval between the two dashed lines can be considered as a neighbourhood in one dimension

Ideal step

The basic idea of our model is to consider the position change of a particle as the result of a mixture of external forces induced by other simulation objects within a certain neighbourhood of this particle. In most cases, real pedestrians make their decisions spontaneously only with incomplete information of the whole situation, that is, the objects in their presence. Therefore, we choose neighbourhoods with a relatively small radius in our model, and this reduces in addition computational complexity. On the other hand, we believe that in a single simulation step with a length of 1~2 s (in our model Δt=1 s), it is impossible for the pedestrians to obtain an overall and exact knowledge of the whole situation. They are guided primarily toward their destinations (the exits in the simulation model) and by additional indicators.

In the sequel, a particle's ideal step will be written as $\Delta\boldsymbol{p}$, and by $\overline{\Delta}\boldsymbol{p}$ we denote the actual position change of this particle in the simulation. For $n\geq 1$, a particle with an initial velocity of $\overline{\boldsymbol{v}}^{(n-1)}$ resulting from the last simulation step experiences an ideal position change $\Delta^{(n)}\boldsymbol{p}$ governed by

$$\Delta^{(n)}\boldsymbol{p}=\left\|\overline{\boldsymbol{v}}^{(n-1)}\right\|\cdot\boldsymbol{n}^{(n)}\cdot\Delta t+\boldsymbol{a}^{(n)}\cdot(\Delta t)^2/2 \tag{7}$$

and

$$\boldsymbol{v}^{(n)}=\left\|\overline{\boldsymbol{v}}^{(n-1)}\right\|\cdot\boldsymbol{n}^{(n)}+\boldsymbol{a}^{(n)}\cdot\Delta t \tag{8}$$

in a simulation step of length Δt, where the acceleration $\boldsymbol{a}^{(n)}$ is the result of the total force. $\boldsymbol{v}$ and $\overline{\boldsymbol{v}}$ denote the ideal velocity and the actual velocity, respectively. The initial velocity $\left\|\overline{\boldsymbol{v}}^{(0)}\right\|$ (as a scalar) is set as the particle's normal velocity $\overline{v}\cdot r$. The general case of $\overline{\boldsymbol{v}}$ will be discussed in the next subsection. In addition, the length of the actual velocity $\boldsymbol{v}^{(n)}$ must be bounded by the particle's maximum velocity (cf. Eq.(2)). Eq.(7) results from the integration of Eq.(8). $\boldsymbol{n}^{(n)}$ is a local navigation vector in a temporary direction, under which the particle restarts its movement with the velocity (as a scalar) achieved in the last simulation step.

The unit vector $\boldsymbol{n}$ in Eqs.(7) and (8) results from the normalization

$$\boldsymbol{n}=\boldsymbol{N}/\|\boldsymbol{N}\|$$

of a navigation term

$$\boldsymbol{N}=\frac{\boldsymbol{o}_p-\boldsymbol{p}}{\|\boldsymbol{o}_p-\boldsymbol{p}\|}+\frac{c(N)}{d(s)}\cdot(\cos\alpha,\ \sin\alpha).$$

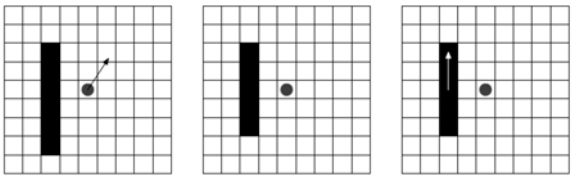


Fig.3 Several repellors within the Moore neighbourhood of a particle with radius 4
(a) The acceleration $\boldsymbol{a}$ induced solely by repellors, calculated later by Eq.(10); (b) y-component of $\boldsymbol{a}$ is zero; (c) An indicator (drawn with a white arrow)

This local navigation term is the sum of two vectors. The first component is the normalized vectorial difference between the exit position and the current position of the particle, which stands for a global flow direction; the second is induced by the closest—The 'closest' indicator can be considered as the maximum norm $\|\cdot\|_\infty$ of all the indicators; alternatively, the 1-norm $\|\cdot\|_1$ (average of all the indicators) is also applicable—indicator $s=(\boldsymbol{s},\alpha)$ (or a randomly selected one in case there exist multiple indicators with the same distance) in the neighbourhood, weighted by the distance $d(s)$ of this indicator and a further constant $c(N)$. For this distance we have

$$d(s)=\sqrt{\left(\frac{(\boldsymbol{s}-\boldsymbol{p})_x}{l_x}\right)^2+\left(\frac{(\boldsymbol{s}-\boldsymbol{p})_y}{l_y}\right)^2},$$

with l_x, l_y being the grid cell length and width defined earlier and $(\boldsymbol{s}-\boldsymbol{p})_x$, $(\boldsymbol{s}-\boldsymbol{p})_y$ being the x-, y-component of the vector $\boldsymbol{s}-\boldsymbol{p}$, respectively.

This constant $c(N)$ is dependent on the neighbourhood N applied in the model,

$$c(N)=\frac{diam(N)}{2}\cdot c_0,$$

where $diam(N)$ denotes the geometric diameter (which is not necessarily twice the radius, in topological sense) of the neighbourhood N, i.e., the largest distance between two points in N. In the case of the Moore neighbourhood Eq.(3), we have

$$c(M_r)=\sqrt{2}r\cdot c_0,$$

and for the von Neumann neighbourhood Eq.(4),

$$c(N_r)=r\cdot c_0,$$

where r is the radius of the neighbourhoods M_r and N_r; c_0 ($0<c_0\leq 1$) reflects the magnitude of the local influence of the indicators. We set $c_0=1$ with the implication that, in a neighbourhood, the 'attraction' of the global exit $\boldsymbol{o}_p$ on the particle p can be fully compensated by an existing indicator in every case. Consequently, the exits only exert a dominant influence on the particles outside the respecting neighbourhoods.

The acceleration $\boldsymbol{a}$ in Eqs.(7) and (8) satisfies

$$\sum \boldsymbol{f}=\boldsymbol{F}=m\cdot\boldsymbol{a},$$

and, derived from Eq.(6), the acceleration term for particle i (positioned at $\boldsymbol{p}_i$) induced by the repulsive force from another particle j (positioned at $\boldsymbol{p}_j$) is therefore

$$\frac{\boldsymbol{f}_{ij}}{m_i}=\frac{Cm_j}{d_{ij}^2}\cdot\boldsymbol{n}_{ij}=\frac{Cm_j(\boldsymbol{p}_i-\boldsymbol{p}_j)}{\left\|\boldsymbol{p}_i-\boldsymbol{p}_j\right\|^3}. \tag{9}$$

Hence, the acceleration $\boldsymbol{a}$ resulting from the repulsive forces on p from other particles and repellors in its neighbourhood can be written as

$$\boldsymbol{a}=c_p\cdot\sum_{p'}\frac{m_{p'}(\boldsymbol{p}-\boldsymbol{p}')}{\left\|\boldsymbol{p}-\boldsymbol{p}'\right\|^3}+c_r\cdot\sum_r\frac{\boldsymbol{p}-\boldsymbol{r}}{\left\|\boldsymbol{p}-\boldsymbol{r}\right\|^3} \tag{10}$$

summing up the impacts induced by all the other particles $p'\in P$ and repellors $r\in R$ in the neighbourhood of p, with proper constants c_p, $c_r>0$ ($c_p=C$ in Eqs.(6) and (9)) parameterizing the 'hardness' of pedestrian and repellor, respectively. Apparently, $\overline{m}\cdot c_p<c_r$ holds, with $\overline{m}$ being the average mass of the particles.

We observe the real world phenomenon of grouping that people tend to be influenced by those with the same (or a similar) destination. For a particle p, let p' be a particle in its neighbourhood. The simplest case is

$$\boldsymbol{o}_p=\boldsymbol{o}_{p'}, \tag{11a}$$

which says that p and p' have the same destination. In Fig.4a we see two particles with similar local directions but the condition Eq.(11a) is not fulfilled. On the other hand, since $p\neq p'$, it is obvious that the area Ω can be divided into Ω_1 and Ω_2 by the straight line through p and p'. We now replace Eq.(11a) by a weaker variant

$$\boldsymbol{o}_p \text{ and } \boldsymbol{o}_{p'} \text{ are in the same subarea } \Omega_1 \text{ and } \Omega_2 \tag{11b}$$

to determine whether p and p' have similar destinations to prompt a grouping effect.

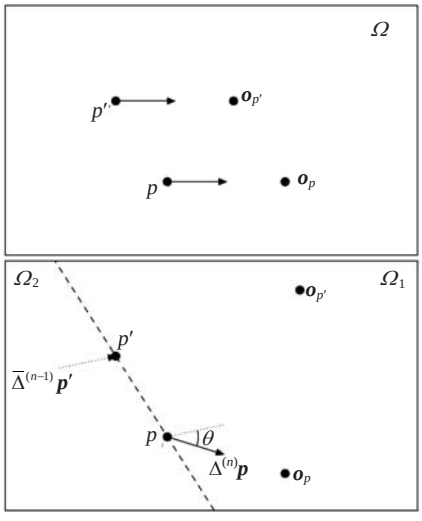


Fig.4 Two examples of grouping effect

(a) p and p' have similar local directions but different destinations; (b) p and p' fulfill the condition Eq.(11b)

Let $n\geq 1$. Let $\overline{\Delta}^{(n-1)}\boldsymbol{p}'$ denote the last actual simulation step of p'. It is reasonable to think that the influence on p depends on the following factors (The superscript '(n)' denoting the simulation step is omitted for simplicity): (1) as usual, $m_{p'}$, the mass of this neighbouring particle p'; (2) the angle $\theta\,(0\leq\theta<\pi)$ between the vectors $\overline{\Delta}^{(n-1)}\boldsymbol{p}'$ and $\Delta^{(n)}\boldsymbol{p}$, i.e., the last actual moving direction of p' and the current ideal moving direction of p; (3) the length of the last actual step $\overline{\Delta}^{(n-1)}\boldsymbol{p}'$ of p', which relates to the temporary velocity of p'; (4) the current distance $\|\boldsymbol{p}-\boldsymbol{p}'\|$ between particles p and p'.

Thus, the grouping effect can be formulated as

$$\boldsymbol{a}_{\mathrm{g}}=c_{\mathrm{g}}\cdot\sum_{p'}\frac{m_{p'}f(\theta)}{\|\boldsymbol{p}-\boldsymbol{p}'\|^2}\cdot\frac{\overline{\Delta}^{(n-1)}\boldsymbol{p}'}{\|\overline{\Delta}^{(n-1)}\boldsymbol{p}'\|} \tag{12}$$

summing up the impacts induced by the particles p' in the neighbourhood fulfilling Eq.(11a) or Eq.(11b), with a certain constant $c_{\mathrm{g}}>0$. $f(\theta)$ can be set as

$$f(\theta)=\begin{cases}\cos\theta, & 0\leq\theta<\pi/2,\\ 0, & \text{otherwise.}\end{cases} \tag{13}$$

The second case in Eq.(13) says that p is not to be affected by p', if the angle θ between their positions is sufficiently large (no less than $\pi/2$, that is to say, the particles have obviously different destinations).

To include the grouping effect in the model, we may simply set

$$\boldsymbol{a}:=\boldsymbol{a}+\boldsymbol{a}_{\mathrm{g}}.$$

Nishinari *et al*.(2004) mentioned a so-called 'inertia effect' claiming that the particles always try to keep their velocities and directions as long as possible. This could be simulated somehow by

$$\boldsymbol{a}_i=\frac{4m_p f(\vartheta)}{\|\boldsymbol{p}^{(n)}-\boldsymbol{p}^{(n-1)}\|^2}\cdot\frac{\overline{\Delta}^{(n-1)}\boldsymbol{p}}{\|\overline{\Delta}^{(n-1)}\boldsymbol{p}\|}, \tag{14}$$

where $c_i>0$ is a constant of similar magnitude as c_{g} in Eq.(12). ϑ denotes the angle between the vectors $\overline{\Delta}^{(n-1)}\boldsymbol{p}$ and $\Delta^{(n)}\boldsymbol{p}$, i.e., the last actual moving direction and the current ideal moving direction of p. A closer look at Eq.(14) reveals that $\|\boldsymbol{p}^{(n)}-\boldsymbol{p}^{(n-1)}\|^2/4$ is the square of the difference between particle p's actual position and its position averaged in the last simulation step, since

$$\frac{1}{2}\|\boldsymbol{p}^{(n)}-\boldsymbol{p}^{(n-1)}\|=\left\|\boldsymbol{p}^{(n)}-\frac{1}{2}(\boldsymbol{p}^{(n)}+\boldsymbol{p}^{(n-1)})\right\|.$$

However, this effect is not included in our model, since in the governing equations of the particle motion Eqs.(7) and (8), acceleration and velocity have already been considered. On the other hand, the time span $\Delta t=1$ s we choose is sufficiently large, in which the pedestrians gather the necessary information from the surroundings, make decisions and move accordingly.

Real step and Bresenham's algorithm

The most studied case $v_{\max}=1$ m/s deals with a position change $\Delta\boldsymbol{p}=\{(1, 0), (-1, 0), (0, 1), (0, -1)\}$. Sometimes it even includes diagonal movements, i.e., further step choices from the set $\{(1, 1), (1, -1), (-1, 1), (-1, -1)\}$. In this case, the particle must cover approximately 41% ($0.41\approx\sqrt{2}-1$) more space than that in a usual step along the x- or y-axis. This manifests a substantial relative difference. In our model, the steps will be approximated on the grid to minimize this relative difference.

Originally, the algorithm of Bresenham for a straight line (Black, 2004) was used for digital plotting. This algorithm assumes that the path of two adjacent points is composed of the mesh points nearest to the desired line segment. Similarly, on the grid Ω, starting with the position $\boldsymbol{p}_1=(x_1, y_1)$ and ending with the position $\boldsymbol{p}_2=(x_2, y_2)$, the cells closest to the moving direction can be determined by the algorithm.

We consider the nontrivial case $\Delta\boldsymbol{p}\neq(0, 0)$ of an ideal step given by Eq.(7), from position $\boldsymbol{p}_1=\boldsymbol{p}^{(n-1)}$ to position $\boldsymbol{p}_2=\boldsymbol{p}^{(n)}$. By Eq.(8), the ideal velocity $\boldsymbol{v}$ is known in addition.

If the particles are dense to a certain level in the simulation system, conflicts among them tend to take place. In this case, there exists a first blocking position $\boldsymbol{p}_{\mathrm{b}}$, and the particle in motion can proceed to position $\boldsymbol{p}_{\mathrm{a}}$ just before being blocked for further

movement (Fig.5). If such a conflict takes place, we reset the velocity on a linear scale from $\mathbf{0}$ to its ideal value $\boldsymbol{v}$, that is,

$$\boldsymbol{v} := \frac{\|\boldsymbol{p}_1\boldsymbol{p}_a\|}{\|\boldsymbol{p}_1\boldsymbol{p}_2\|}\cdot\boldsymbol{v}, \tag{15}$$

in relation to the length of its advancement on the grid. Obviously, the real step of the particle is then $\bar{\Delta}\boldsymbol{p} = \boldsymbol{p}_a - \boldsymbol{p}_1$.

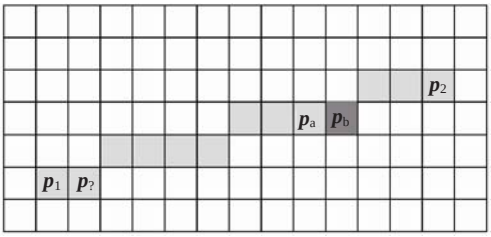


Fig.5 The cells which a particle should go through from position $p_1=(x_1, y_1)$ to $p_2=(x_2, y_2)$ using Bresenham's algorithm for a straight line
The cell at $\boldsymbol{p}_b$ is blocked by some other object

Furthermore, we notice that the velocity of a particle p is bounded by its maximum defined in Eq.(2). Thus for the actual velocity $\bar{\boldsymbol{v}}$, we have

$$\bar{\boldsymbol{v}} = \min\left(\|\boldsymbol{v}\|, \bar{v}\cdot r_p\cdot q\right)\cdot\frac{\boldsymbol{v}}{\|\boldsymbol{v}\|},$$

with $\bar{v}$ given in Eq.(1), r_p in Eq.(2), and q=3 as a constant factor in the model.

Comment 1 Eq.(8) describes an undisturbed velocity change from position $\boldsymbol{p}_1$ to position $\boldsymbol{p}_2$, with the actual velocity in the last simulation step as the lower bound. However, when we take into consideration the possible conflict, we choose to set this lower bound in Eq.(15) as $\mathbf{0}$. Considering the case of $\boldsymbol{p}_?$ as a blocking position in Fig.5, the particle is fully blocked in the current step. It is understandable that the new velocity becomes $\mathbf{0}$.

Comment 2 In the implementation (for Bresenham's algorithm, see Appendix), the quotient $\|\boldsymbol{p}_1\boldsymbol{p}_a\|/\|\boldsymbol{p}_1\boldsymbol{p}_2\|$ in Eq.(15) can be replaced by $(i-1)/dx$.

Comment 3 A hardware-near implementation of a Bresenham's algorithm can be undertaken for a given v_{max} by saving up the pre-computed paths as lists of the grid points for all possible $\Delta\boldsymbol{p}$'s.

Comment 4 In the implementation, the particles may be given a further choice (but with a proper probability) of a small 'random walk', i.e., the movement toward one of the direct neighbouring and accessible cells, when it is blocked by other particles.

Update scheme

Two popular update schemes are 'parallel update' and 'random sequential update' (Keßel *et al*., 2002). In the parallel update, the particles undertake a 'virtual' step at the same time. In the case of a multiple occupation of a single cell, additional mechanisms are to be applied to select a 'winner' from the virtually occupying particles. All the 'loser' particles stay at their original positions. In the random sequential update, probabilities of all potential steps for all particles will be calculated based on the static and dynamic floor fields and, accordingly, the particles' step choices can be made (Burstedde *et al*., 2001; Schadschneider, 2002; Nishinari *et al*., 2004).

In our model, we sort the particles into a sequence by their velocities and the particles will be processed in this sequence. This has the same effect as 'the quicker one makes the move first'. Another choice is to associate every particle p with a dynamic distance value, e.g., the local distance to its destination cell at $\boldsymbol{o}_p$. After every simulation step, all the particles can be sorted by this distance value.

In case of a velocity larger than one cell per step, it would be theoretically necessary to verify the accessibility of all the cells along the path, because these cells can be on the computed paths of other particles in the same simulation step. This can be computationally unfavourable. In applying our sequential update, we check the accessibility of these cells according to the computation of the last step. If a parallel update is applied, the small possibility that there could be additional blocking points on the path because of the state change of some cells in the same simulation step cannot be fully excluded. This possibility becomes noticeable for a large v_{max}. On the other hand, extreme v_{max} values would be unrealistic for the model, considering the geometric construction of the grid (the reader may recall that a velocity of five cells per step corresponds to a walking speed of 2.5 m/s and r_p=5/3).

Parameter choice

In Eq.(5), the social force model (Helbing *et al*., 2000a) was configured as A_i=2000 N, B_i=0.08 m, and r_{ij}=0.6 m (as the mathematical expectation of a uniform distribution). $\overline{m}$, the average mass of the pedestrians, was assumed to be 80 kg. In this subsection we sometimes leave out the units of measurement for simplicity. Assuming $m_i = m_j = \overline{m}$ for simplicity, we consider the function

$$C(d) = \frac{d^2}{80^2} \cdot 2000\,\mathrm{e}^{(0.6-d)/0.08} \quad \text{for } d>0.6, \qquad (16)$$

and the integrals

$$S(d; a, b) = \int_a^b C(d) \cdot \frac{80^2}{x^2}\mathrm{d}x \quad \text{for } 0<a<b,$$

$$S_{\mathrm{SFM}}(a) = \int_a^{+\infty} 2000\mathrm{e}^{(0.6-x)/0.08}\mathrm{d}x \quad \text{for } a\geq 0.6,$$

as the total sums in the 1D case. As for Eq.(16), Eqs.(6) and (5) share the same value locally at d (letting C=$C(d)$); d is required to be larger than 0.6, since Eq.(5) requests that the two pedestrians do not touch each other, i.e., $d>r_{ij}$. After simplification, we have

$$S(d; a, b) = 80^2 \cdot C(d) \cdot [-1/x]\Big|_{x=a}^{x=b}, \qquad (17)$$

$$S_{\mathrm{SFM}}(a) = 160\,\mathrm{e}^{7.5-12.5a}. \qquad (18)$$

In Eq.(18), a is the lower bound of the distance of two particles for which the interaction will be considered. $S_{\mathrm{SFM}}(a)$ decreases at a considerable rate for small a: $S_{\mathrm{SFM}}(0.6)$=160, $S_{\mathrm{SFM}}(0.8)\approx$13.1, $S_{\mathrm{SFM}}(1)\approx$1.08, $S_{\mathrm{SFM}}(1.5)\approx$0.002 08, and so on. Our impression is that A_i might have been slightly too large and B_i too small; on the other hand, it can be argued that the model of (Helbing *et al*., 2000a) was used for evacuation scenarios of high particle densities so that the interactions among particles within small distances should be given preferential consideration. In Eq.(17), a and b characterize the neighbourhood in the 1D case; a typical choice would be a=0.5 (one cell size in 1D, unit of measurement omitted), b=10a, and thus $[-1/x]\Big|_{x=a}^{x=b}$=1.8 (The contribution of b is relatively small, e.g., for b=5a, $[-1/x]\Big|_{x=a}^{x=b}$ turns out to be 1.6). Combining Eq.(17) with Eq.(16), we have

$$\frac{S(d; 0.5, 5)}{80^2 \cdot 1.8} = C(d) = \frac{d^2}{80^2} \cdot 2000\,\mathrm{e}^{(0.6-d)/0.08}.$$

We recall that the solution of this equation is where Eqs.(6) and (5) share the same function value locally (Fig.2). We list some possible configurations for different $S(d; 0.5, 5)$ values in Table 2.

Table 2 Some possible parameter configurations

$S(d; 0.5, 5)$	$C(d)$	Numerical solution for d
10	0.000 868	1.080
25	0.002 170	0.997
40	0.003 470	0.952
50	0.004 340	0.931
75	0.006 510	0.891
100	0.008 680	0.863

For d=1 m [the original configuration of (Helbing *et al*., 2000a)], $C(d)\approx$0.002 N·m^2/kg^2; at the same time, $C(d)$=0.004 N·m^2/kg^2 corresponds to an $S(d; 0.5, 5)$ somewhere in the interval (40, 50) and d is roughly 0.94 m. [0.002, 0.004] seems to be a reasonable range for c_p=C=$C(d)$. Some of the test results of different configurations are given in Table 3 and Fig.6.

Table 3 Results from some test cases

Case	Radius r	c_p	$c_r / \overline{m}$	c_g	n_{ped} (final)
(a)	0	–	–	–	360
(b)	3	0.002	0.006	0.001	377
(c)	3	0.002	0.006	0.001	112
(d)	3	0.004	0.008	0.002	79
(e)	3	0.003	0.010	0.001	109
(f)	5	0.003	0.010	0.001	211

In test case (a), since the von Neumann neighbourhood radius r was set to be zero, the constants c_p, c_r and c_g became irrelevant for the simulation, and the overall behaviour tended to be chaotic—the somehow 'blind' behaviours of the particles induced a high particle density in the simulation and this caused further conflicts among the particles and a higher particle density in return. In test case (b), the neighbourhood radius was set as 3, and the repulsive effect of the repellors can be easily identified.

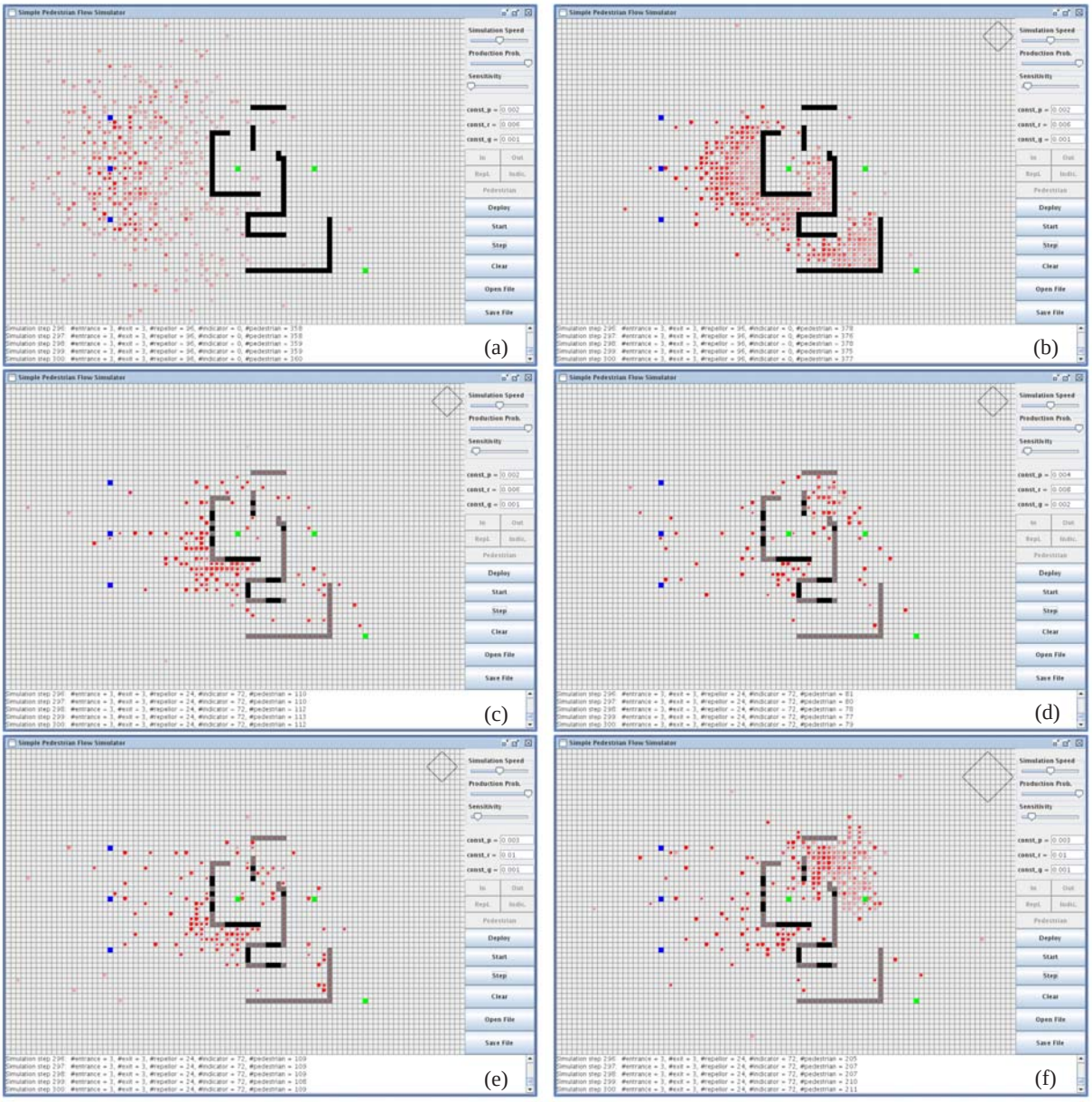


Fig.6 Screenshots of the test cases in Table 3 of three entrances and three exits

Entrances (in dark grey), exits (in light grey) and repellors (in black) are drawn in rectangular shape. Indicators are drawn in grey with a small line segment giving the direction. Cases (a) and (b) have the identical geometry; the same is for (c), (d), (e) and (f). The second geometry (cases (c)~(f)) differs from the first one (cases (a) and (b)) in that some of the repellors are replaced by indicators. Particles, drawn in filled circles, were produced at the entrances and associated with a randomly selected exit. The contour of the neighbourhood (when the neighbourhood radius is larger than 0) can be identified on the corner. All six simulation tests ran for 300 steps

Test cases (c), (d) and (e) were configured with the same neighbourhood radius but slightly different c_p, c_r and c_g constants. In these three test cases, the simulation ran smoothly (in the sense of particle density, which is the balanced result of flow-in at the entrances and flow-out at the exits). Test case (d) exhibited the lowest particle density, and the particles in the simulation were generally closer to the exits than in other test cases. A larger von Neumann neighbourhood radius of 5 caused congestion in test case (f). A more refined parameter calibration may be possible when the simulation is tested on a global level and compared with statistical data.

FURTHER DISCUSSION

Since we have dealt with the general case $v_{max}\geq$ 1 m/s, the refinement of the grid Ω becomes possible. The other general case $v_{max}<1$ m/s would not be interesting, since this implies larger grid cells, and consequently the minimum space that a real pedestrian reserves exclusively for himself/herself is not realistic and therefore meaningless. Besides the neighbourhood Eq.(3) or Eq.(4) for objects of all types, we may introduce a second neighbourhood of a smaller radius r', in which only particles will be counted. Among the particles a new kind of repulsive force can be defined. This has the advantage that, when we extend our model to simulate evacuation or emergency situations, we may encounter the fact that pedestrians are sometimes forced to stay much closer to each other than in the normal case. A more refined grid Ω' enables multiple particles to be put into a single cell in the original grid Ω and yet in this special case the interactions among the particles in the same cell in Ω can be considered and evaluated on the grid Ω'.

We also wish to point out that choosing a large radius for the neighbourhood may not be proper. This causes a quadratic increase in the computation and, on the other hand, a real pedestrian only takes into consideration the visible objects in the surroundings; this means that the visibility (i.e., the accessibility from the original particle) of the objects in the neighbourhood must be checked first before their influences on that particle can be computed, which is again very demanding in respect of computational complexity.

When we use the model for evacuation, Eq.(2) may be extended to the form like

$$p=(m_p, \boldsymbol{p}, O_p, r_p),$$

with m_p, $\boldsymbol{p}$ and r_p unchanged, but equipped with an additional set

$$O_p \subseteq O\times\mathbb{R}$$

of all possible exits and their priority values. These priorities should be dynamically evaluated, so that an exit at $\boldsymbol{o}_p$ with the highest priority can always be determined and p can then take the form $(m_p, \boldsymbol{p}, \boldsymbol{o}_p, r_p)$, which fits into our original model. Under certain circumstances, e.g., in the case of an abnormally high particle density at a local position or when a particle takes 'irrational' moves continuously (which can be detected when the distance value toward the destination mentioned in the subsection of "Update scheme" does not diminish after a sufficiently long time period), other exits may be tried. This concerns navigation on a global level and goes beyond the realm of the current text.

References

Black, P.E., 2004. Bresenham's Algorithm. Dictionary of Algorithms and Data Structures. U.S. National Institute of Standards and Technology. Http://www.nist.gov/dads

Burstedde, C., Klauck, K., Schadschneider, A., Zittartz, J., 2001. Simulation of pedestrian dynamics using a two-dimensional cellular automaton. *Phys. A*, **295**(3-4): 507-525. [doi:10.1016/S0378-4371(01)00141-8]

Chopard, B., Masselot, A., 1999. Cellular automata and lattice Boltzmann methods: a new approach to computational fluid dynamics and particle transport. *Fut. Gen. Comput. Syst.*, **16**(2-3):249-257. [doi:10.1016/S0167-739X(99)00050-3]

Esser, J., Schreckenberg, M., 1997. Microscopic simulation of urban traffic based on cellular automata. *Int. J. Mod. Phys. C*, **8**(5):1025-1036. [doi:10.1142/S0129183197000904]

Helbing, D., 1997. Verkehrsdynamik: Neue Physik-alische Modellierungskonzepte. Springer-Verlag Berlin Heidelberg (in German).

Helbing, D., Farkas, I., Vicsek, T., 2000a. Simulating dynamical features of escape panic. *Nature*, **407**(6803): 487-490. [doi:10.1038/35035023]

Helbing, D., Herrmann, H.J., Schreckenberg, M., Wolf, D.E. (Eds.), 2000b. Traffic and Granular Flow'99. Springer-Verlag Berlin Heidelberg.

Hoogendoorn, S.P., Luding, S., Bovy, P.H.L., Schreckenberg, M., Wolf, D.E. (Eds.), 2005. Traffic and Granular Flow'03. Springer-Verlag Berlin Heidelberg.

Keßel, A., Klüpfel, H., Wahle, J., Schreckenberg, M., 2002. Microscopic Simulation of Pedestrian Crowd Motion. *In*: Schreckenberg, M., Sharma, S.D. (Eds.), Pedestrian and Evacuation Dynamics. Springer-Verlag Berlin Heidelberg, p.193-200.

Klüpfel, H.L., 2003. A Cellular Automaton Model for Crowd Movement and Egress Simulation. PhD Thesis, Universität Duisburg-Essen, North Rhine-Westphalia, Germany. Http://www.ub.uniduisburg.de/ETD-db/theses/available/duett-08012003-092540/

Nagel, K., Schreckenberg, M., 1992. A cellular automaton model for freeway traffic. *J. Phys. I Fr.*, **2**:2221-2229. [doi:10.1051/jp1:1992277]

Nishinari, K., Kirchner, A., Namazi, A., Schadschneider, A., 2004. Extended floor field CA model for evacuation dynamics. *IEICE Trans. Inf. Syst.*, **E87-D**(3):726-732.

Palmer, A., Bailey, R., 1975. Sex differences and the statistics of crowd fluids. *Behav. Sci.*, **20**(4):223-227. [doi:10.

1002/bs.3830200403]
Predtetschenski, W.M., Milinski, A.I., 1971. Personenströme in Gebäuden: Berechnungs-methoden für die Projektierung. Staatsverlag der Deutschen Demokratischen Republik, Leipzig, DDR (translated into German from Russian).
Schadschneider, A., 2002. Cellular Automaton Approach to Pedestrian Dynamics—Theory. *In*: Schreckenberg, M., Sharma, S.D. (Eds.), Pedestrian and Evacuation Dynamics. Springer-Verlag Berlin Heidelberg, p.75-85.
Schadschneider, A., Schreckenberg, M., 1993. Cellular automaton models and traffic flow. *J. Phys. A. Math. Gen.*, **26**(15):L679-L683. [doi:10.1088/0305-4470/26/15/011]
Schadschneider, A., Pöschel, T., Kühne, R., Schreckenberg, M., Wolf, D.E. (Eds.), 2007. Traffic and Granular Flow'05. Springer-Verlag Berlin Heidelberg. [doi:10.1007/978-3-540-47641-2]
Schreckenberg, M., Sharma, S.D. (Eds.), 2002. Pedestrian and Evacuation Dynamics. Springer-Verlag Berlin Heidelberg.
Schreckenberg, M., Wolf, D.E. (Eds.), 1998. Traffic and Granular Flow'97. Springer-Verlag Singapore.
Waldau, N., Gattermann, P., Knoflacher, H., Schreckenberg, M. (Eds.), 2007. Pedestrian and Evacuation Dynamics 2005. Springer-Verlag Berlin Heidelberg. [doi:10.1007/978-3-540-47064-9]

APPENDIX: BRESENHAM'S ALGORITHM

According to (Black, 2004), the path of two grid points is composed of the mesh points nearest to the desired line (or curve) segment. In octant I, a revised version of the code reads

```
/* Bresenham's algorithm
 * for straight line */
void bresenham_line_oct1(int x1, int y1, int x2, int y2)
{
        int x=x1;
        int y=y1;
        int dx=x2−x1;
        int dy=y2−y1;
        int dx2=2*dx;
        int dy2=2*dy;
        int e=dy2−dx;

        for (int i=1; i<=dx; i++)
        {
                move_to(x, y);
                if (e>=0)
                {
                        y++;
                        e−=dx2;
                }
                x++;
                e+=dy2;
        }
        move_to(x, y);
}
```

In octants II, III, VI and VII, the loop variable *i* corresponds to the *y*-direction. Small changes concerning *e* and the loop body will be necessary to ensure the correct proceeding from the start point to the end point.