Mathematical modeling of pedestrian flows using cellular automata and dynamic stepsizing

Günter Bärwolff, Minjie Chen and Hartmut Schwandt

Technische Universität Berlin, Institut für Mathematik

Abstract. We present a two-dimensional automaton model to simulate pedestrian flows. In this model, pedestrians may be allocated have preassigned or randomly chosen starting points and destinations, and they may get influenced by additional repelling or indicating factors. Interior walls and other obstacles can be abstracted as repellors. We apply Bresenham's algorithm [1] of line rastering to calculate the ideal forward step which a single pedestrian may take on a two-dimensional grid. We introduce a flexible choice of step sizes to improve the basic algorithm.

Keywords: Cellular automaton; traffic simulation; collision; obstacle handling
PACS: 45.70.Vn, 45.50.Dd, 45.70.-n, 47.80.Jk, 45.50.Tn

INTRODUCTION

In the pioneering work [2] a one-dimensional cellular automaton was proposed to simulate traffic flows in a circled closed system. In the two-dimensional case, [3] introduced a driven random walk model in which the pedestrian, in the following called *particle*, has in a single simulation step may take basically one out of three choices to go forward, up or down on a two-dimensional grid with the possibilities

$$P(\text{forward}) = D + \frac{1-D}{\#\text{unoccupied cells}}\ ,\ P(\text{upward}) = \frac{1-D}{\#\text{unoccupied cells}}\ ,\ P(\text{downward}) = \frac{1-D}{\#\text{unoccupied cells}},$$

where $D \in [0,1]$ is the so-called drifting constant. This implies
and demands a normalisation in accordance with $\sum P$.

Combining this with the new idea of "floor fields", [4] and [5] offered new ansatzes focusing on the case of $v_{\max} = 1$,[1] i.e. , the possible moves that a particle at the position (c_x, c_y) may take are confined to be towards its immediate neighbours at $(c_x + \Delta x, c_y + \Delta y)$ for $\Delta x, \Delta y \in \{-1, 0, 1\}$.

We would like to consider the case $v_{\max} > 1$. Of specific interest are strategies for conflict resolution of the particle movements that also give adequate consideration to computational efficiency.

MODEL

In our model, the geometry is defined on a two-dimensional 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 number of the cells on the x-, y-axis respectively. From a sociological

# unoccupied cells	$P($**forward**$)$	$P($**upward**$)$, $P($**downward**$)$	$\sum P$
1	1	$1-D$	$3-2D$
2	$\frac{1}{2}+\frac{D}{2}$	$\frac{1}{2}-\frac{D}{2}$	$\frac{3}{2}-\frac{D}{2}$
3	$\frac{1}{3}+\frac{2D}{3}$	$\frac{1}{3}-\frac{D}{3}$	1

[1] Mathematically, this should be written as $v_{\max} \cdot \Delta t =$ cell length (width), where Δt denotes the time span of each simulation step. In this text, cell length and width are scaled to be 1.

perspective, $l_x = l_y = 0.7$m might be a proper choice for persons from western cultures, whereas in the oriental cultures, smaller l_x, l_y values would be accepted or even preferred. Scaled by l_x and l_y, this can be written in discrete form, $\Omega = \{1,\dots,n_x\} \times \{1,\dots,n_y\}$. The grid is not confined to be of rectangular shape, when the cells adjacent to the boundary are of the repellor type, will be discussed in the next section. Our assumtions are:

1. The contents of the cells will be called *object*s. An empty cell is of the object type "empty".
2. The pedestrians, i.e. the particles, may differ in terms of their behavior. Their individual characters can be controlled by various parameters. In general, this applies to all the non-empty objects.
3. The particles possess no a priori knowledge of how to reach a local optimum. A single particle aiming at a fixed exit gathers the information at each simulation of the best choice of step at each simulation step.
4. The particles are neither "cooperative" (to reach an overall optimum) nor "competitive" (so that the overall optimisation would be hindered by local optima).

With the characterisation of "Cell", "Object" and "Neighbourhood", the general aim of particles and the definition of rules in the case of collisions and obstacles (confilcts) we can evaluate the actual flow direction and realise steps.

ALGORITHM, STEP CHOICE AND CONFLICT RESOLUTION

In the most studied case $v_{\max} = 1$, we see, as shown in the left of Figure 1, that a particle at the position (c_x, c_y) may access the cells (c_x+1, c_y) and (c_x, c_y+1) in one step. This particle may also access the upper right cell located at (c_x+1, c_y+1) in one step. This implies that in doing so the particle must cover 41% more space than that in a usual step along the *x*- or *y*- axis.

Bresenham's Algorithm for Circles

In our model for the generalised case $v_{\max} \geq 1$, we denote the time lapse of a single update by Δt and the local velocity of a particle p by v_p. Thus, at the position (c_x, c_y), the particle could reach any cell with a distance $r_p = v_p \cdot \Delta t$. We call the circle[2]

$$(x - c_x)^2 + (y - c_y)^2 = r_p^2 \tag{1}$$

the accessibility circle of p at (c_x, c_y). In the discrete case, this circle can be approximated on the grid Ω by a series of cells (partly drawn in gray on the right of Figure 1) using the circle variant of Bresenham's algorithm. If an ideal move had been computed, which however becomes impossible due to conflict with others, the particle p still has alternatives. Theorectically, going diagonally to the right or going back is also an option.

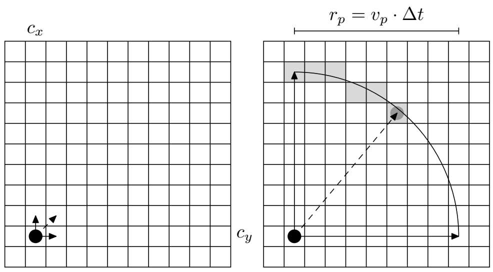

FIGURE 1. Possible steps for a single particle. Left: Up to the right or diagonally to the right. Right: An example of Bresenham's algorithm in the first and second octants. The particle may have a relatively larger step width, the cells within reach form an arc. This arc can be computed by the circle variant of Bresenham's algorithm.

Note that in the case $v_{\max} = 1$, $(c_x \pm 1, c_y)$ and $(c_x, c_y \pm 1)$ the four grid points are computed by the circle variant of Bresenham's algorithm. The remaining four grid points $(c_x \pm 1, c_y \pm 1)$ and $(c_x \mp 1, c_y \pm 1)$ might also be taken into consideration, since $\frac{\sqrt{2}}{2} < 1 < \sqrt{2}$.

---

[2] Obviously in (1), x, y, c_x, c_y and r_p are to be taken as float value before the scaling by cell length and width, because the integer solution for (1) is not guaranteed.

The problem of step choice is, in a broarder sense, the choice of an accessible cell using the circle variant of Bresenham's algorithm. Considering the rational behaviour of human beings — and since after computing the **v** vector, we know the ideal moving direction — we approximate this in two partial stages:

1. search for the first blocking cell using Bresenham's algorithm for straight line (see Figure 2 below);
2. make small modifications of the cell position which has been successfully accessed last.

Bresenham's Algorithm for Straight Line

Originally, the famous algorithm of J. E. Bresenham [1] 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 at cell $p_1 = p_{\text{pos}}^{(n)}$ and ending at cell $p_2 = p_{\text{pos}}^{(n+1)}$ with $p_1 = (x_1, x_2)$ and $p_2 = (x_2, y_2)$ respectively, the cells closest to the moving direction can be computed by the algorithm.

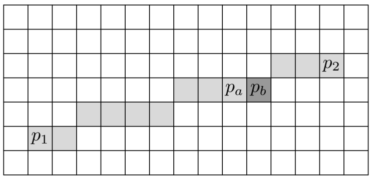

FIGURE 2. The cells a particle should go through from position $p_1 = (x_1, y_1)$ to $p_2 = (x_2, y_2)$ using Bresenham's algorithm for straight line. The cell at p_b is blocked by some other object.

If the particles are to a certain level in the simulation system dense, conflicts tend to take place more frequently for a particle in the way from its original position $p_1 = (x_1, y_1)$ to the ideal new position (x_2, y_2) while holding (1) (see Figure 1). In this case, we first apply Bresenham's algorithm for straight line to determine all the cells the particle is expected to go through before reaching $p_2 = (x, y)$. If an object positioned at cell p_b (see Figure 2) blocks the way, we try a small modification based on the cell last accessed $p_a = (x_a, y_a)$. Again, theorectically, a second accessibility circle

$$(x - x_a)^2 + (y - y_a)^2 = (r_p - \| p_a - p_1 \|)^2$$

can be built and a further choice can be made.

Example of Conflict Resolutions

The following choices may be applied to resolve the conflicts.

1. The particle stops (as at position p_a Figure 2). This is an acceptable choice when the time lapse of each simulation step Δt is small, we may then take for granted that the particle does not have enough time for a further decision.
2. The particle may swerve to the right, given that the neighboring cell is not blocked. This reflects human social behaviour. A simple example is presented in Figure 3.
3. Try a second accessibility circle. In the implementation, we chose a small radius in this second sub-step, $r_p = 1$, with the implication that the particle p suffers a slight time loss due to the blocked cell at p_b. This enables us to choose one of four grid points $(c_x \pm 1, c_y)$ and $(c_x, c_y \pm 1)$ as a solution. Furthermore, as mentioned above, the four corner points $(c_x \pm 1, c_y \pm 1)$ and $(c_x \mp 1, c_y \pm 1)$ might also be considered. The choice may be done on a random basis, or, slightly more sophisticated, the better the ideal moving direction can be kept by a candidate point, the more should this point be preferred. In Figure 1, when the ideal moving direction is shown as that on the right, the cell at $(c_x + 1, c_y + 1)$ should be preferred as an solution.

4. Finally, if no solution is available, the particle has to remain at the cell last accessed (p_a in Figure 2).

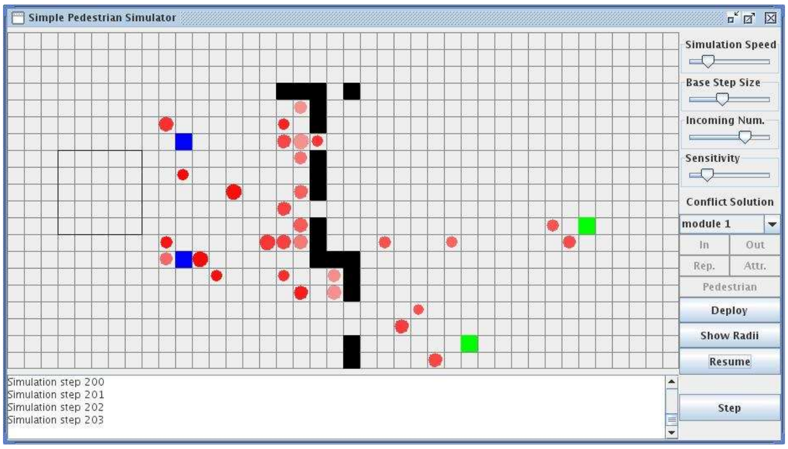

FIGURE 3. Screenshot of a simple test case. Blue, green and black cells represent entrances, exits and repellors respectively. Particles, drawn in red, are located randomly at the entrances and aim to go to their destined exits.

CONCLUSION

A discrete pedestrian flow model for a multi-aim situation was constructed and implemented (s. also [6]). Interior walls and other obstacles were considered as repellors. Inhomogeneities of the pedestrians are possible. The local velocities of single pedestrians may differ in their velocity or in their behavior from each other, thus bearing the physical characteristic of a random distribution of pedestrians. Experiments and comparisons of simulations with the developed model are in preparation.

ACKNOWLEDGMENTS

This research project is supported by the Berlin Government and by the European Union (PROFIT program). Further, we would like to thank our colleague Prof. K. Nagel from the Technische Universität Berlin for useful discussions.

REFERENCES

1. P. E. Black: *Bresenham's algorithm*, in: Dictionary of Algorithms and Data Structures, ed. P. E. Black, U.S. National Institute of Standards and Technology (2004); J. E. Bresenham's original paper *Algorithm for computer control of a digital plotter* (1962) is online available at `http://www.research.ibm.com/journal/sj/041/ibmsjIVRIC.pdf`
2. K. Nagel and M. Schreckenberg: *A cellular automaton model for freeway traffic*, J. Phys. I France 2 (1992)
3. A. Keßel, H. Klüpfel, J. Wahle and M. Schreckenberg: *Microscopic simulation of pedestrian crowd motion*, in: Pedestrian and Evacuation Dynamics, eds. M. Schreckenberg and S. D. Sharma, Springer-Verlag, Berlin (2002)
4. C. Burstedde, K. Klauck, A. Schadschneider and J. Zittartz: *Simulation of pedestrian dynamics using a two-dimensional cellular automaton*, Physica A 295 (2001)
5. K. Nishinari, A. Kirchner, A. Namazi and A. Schadschneider: *Extended floor field CA model for evacuation dynamics*, IEICE Trans. Inf. & Syst. Vol. E87-D (2004)
6. G. Bärwolff, M.-J. Chen, H. Schwandt and T. Slawig: *Simulation of Pedestrian Flows for Traffic Control Systems*, Proc. of the Sixth International Conference on Information ans Management Sciences, July 1-6, 2007 Lhasa/Tibet, in Series of Information & Management Sciences of the Cal Poly State University (2007)