

Optimizing Pedestrian Environments with Evolutionary Strategies

Marijn Swenne and Thomas Bäck

Abstract The environment of pedestrians has a big influence on their movement efficiency. Poorly designed environments can lead to high pedestrian densities which in turn increases the risk of injury, or in the case of egress, increases tardiness. A common approach is to have an expert design an environment and test it with a simulator. We propose a method to design a pedestrian environment by using pedestrian simulation in combination with optimization by means of evolutionary strategies. The results demonstrate that optimized environment layouts can improve pedestrian throughput in comparison with basic layouts designed by humans.

Keywords Pedestrian • Environment • Optimization • Evolutionary strategy • Simulator

1 Introduction

The goal of this research is two-fold: First, a realistic pedestrian simulator is proposed, which secondly is used as a fitness evaluator for an Evolutionary Strategy (ES) for optimization. It is demonstrated in this paper how an ES can be used to find an environment that maximizes pedestrian efficiency.

For the simulator, the Social Forces Model (SFM) is used. This model represents one of the three main classes of microscopic pedestrian models. The other ones are the Benefit Cost Cellular Model (BCCM), and the Magnetic Force Model (MFM). The BCCM was first proposed by Gipps and Marksjo [3] in 1985. In this model, the space in which agents are placed is represented as a 2D cellular grid. All agents, obstacles and other objects are defined as a cell or collection of cells. This

M. Swenne (✉) • T. Bäck
Leiden University, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
e-mail: mswenne@liacs.nl; marijn.swenne@gmail.com; baeck@liacs.nl

implies that agent location and movement are discreet while pedestrian location and movement are not. Other well known examples of BCCMs are by Kirchner and Schadschneider [7] and Burstedde et al. [2]. The MFM as proposed by Okazaki [9] is based on each pedestrian, obstacle and goal having a magnetic polarity. The agents and obstacles have a positive pole whereas the agent goals have negative poles. The movement of each agent is the equation of all magnetic fields resulting in agents eventually moving to their goals while avoiding collisions. Finally, the SFM by Helbing and Molnár [6] is very similar to the MFM but assumes that repulsion and attraction are a result of social pressures. SFM is currently one of the most commonly used models.

In the remainder of this paper, the underlying principles of pedestrian behavior are outlined in Sect. 2, and its corresponding simulator implementation is explained in Sect. 3. Section 4 presents the setup of the environment optimization experiments conducted, and Sect. 5 provides a discussion of the results achieved by optimization. Finally, conclusions are given in Sect. 6.

2 Pedestrian Behaviour

To create a realistic simulator we have looked at different empirical studies on various aspects of pedestrian behaviour. Decisions pedestrians make on an individual level are used to model agents. Patterns that occur as a result of these decisions are used to validate the model. Two key components of pedestrian behavior modeling are perception and emergent behavior, as outlined in the following sections.

2.1 Perception

Pedestrians observe their surrounding mainly by sight. Goffman [4] observed pedestrians and developed a model describing how pedestrians take note of the fellow pedestrians. He called this process scanning (see Fig. 1a). In this model only a subset of the other pedestrians are influential. Whether a pedestrian is influential is determined by its proximity, its radial orientation and whether they are separated by other pedestrians.

Pedestrians only take note of what is in front of them and within their influence area. This influence area is shaped like a halved ellipse stretched in front of the pedestrian. Its size is 2 m in radius on the minor axis aligned to the width of the pedestrian and 5 m in radius on the major axis in front of the pedestrian. Also, other pedestrians will only be of influence if they are in direct line of sight.

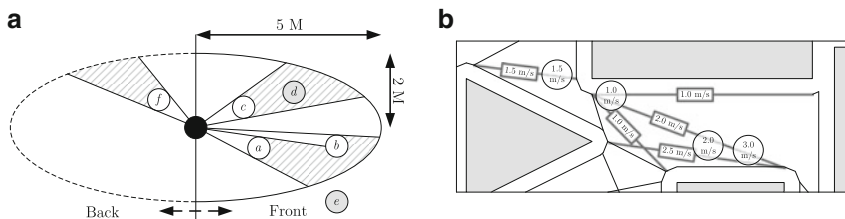


Fig. 1 Agent perception in the social forces model and path selection. **(a)** Goffman scanning. **(b)** Agent speed (*circle*) and average speed on edge (*rectangle*)

2.2 Emergent Behaviour

Emergent behaviour is any behaviour of a system that is not a property of any of the components of that system, but rather results from the interactions of its components. In a pedestrian simulation this would mean any behaviour that is not a property of a pedestrian but the result of inter-pedestrian behaviour. A good example is the pedestrian lanes that emerge at pedestrian stream intersections. Pedestrians do not consciously create them or are even aware of these lanes.

According to Helbing et al. [5] there are three types of emergent behaviour among pedestrians, namely, lane formation, circular flows and the clogging effect. Lane formation happens when two opposing streams of pedestrians interact. This results in pedestrians forming ribbons to avoid collision. Circular flows occur at an intersection where pedestrians enter from multiple directions. Circular flows will occur in the centre of this intersection but have a short lifespan. Finally, the clogging effect occurs at bottlenecks. Groups of pedestrians will form in front of the bottleneck in the shape of half a circle.

3 Pedestrian Simulation Implementation

The simulator developed for this research is based on the social forces model. Functionality for dynamic obstacle avoidance based on the observational studies described in Sect. 2 is implemented in the simulator. The way agents plan their route makes use of a Reduced Visibility Graph (RVG) as described by LaValle [8].

3.1 Pathfinding

Pathfinding ties into the social forces model by influencing the steering force f^γ . It can be separated into three components: Graph construction, path selection and

attractor point selection. For every environment a graph G is constructed. For every agent, one path P is selected from graph G . And for every time step, per agent, one attractor point ap on path P is selected. The following paragraphs give a brief explanation of the implementation of these components.

3.1.1 Graph Construction

First, an Euclidean pathfinding graph G is constructed as follows: Let \mathcal{B} be a collection of obstacle borders, each $b \in \mathcal{B}$ belonging to an obstacle $o \in \mathcal{O}$. b_o is an extension of polygon o by range r , drawing a region of constant width around o .

A common value for r is the radius of an agent. If two obstacle borders are overlapping, the overlapping edges are broken at the point they cut each other. All edges that are enclosed are discarded with the remaining edges forming a new obstacle border, replacing the original two. G is constructed by creating a reduced visibility graph from the polygons contained in \mathcal{B} .

3.1.2 Path Selection

Every edge of G contains congestion data, being the average speed of agents that walk over that edge. An agent is considered to be walking over an edge if the circle representing the agent overlaps with the edge, see Fig. 1b. When calculating a route, it is the expected travel time that determines the shortest path. When congestion increases there is an increased chance of recalculating the route for a possible faster alternative. The result of this path selection is a series of line segments $P = p^n, \dots, p^0$, with p^0 being the agent's goal.

3.1.3 Attractor Point Selection

The attractor point ap is the point that will be used to determine the steering force for agent a . Let T_a be a collection of all tangent lines on obstacles \mathcal{O} that have a 90° angle with agent a and let T'_a be the collection of line segments resulting from breaking each line from T_a where they are overlapping. Attractor region R_a is a polygon constructed from all line segments in T'_a that are in direct line of sight of a and not obstructed by other line segments from T'_a . ap_a is the point where the border of R_a cuts P_a or p_a^0 if no such point exists. See Fig. 2b.

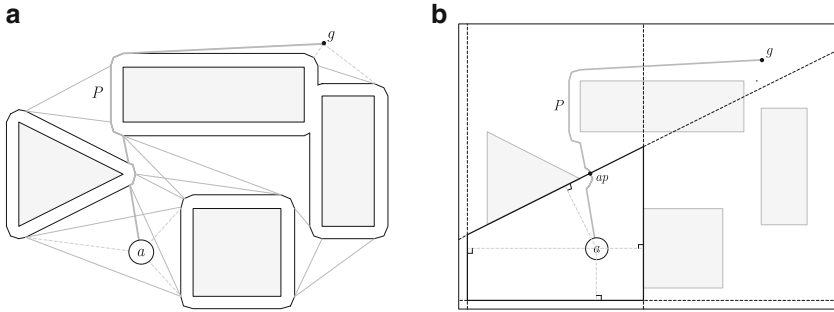


Fig. 2 Pathfinding. (a) Path selection on RVG for agent a to goal g . (b) Attractor box construction and attractor point ap selection

3.2 Social Forces Implementation

In the social forces model \vec{F}_a and \vec{f}_a are the forces resulting from all static obstacles (\vec{F}_a) and all dynamic obstacles (\vec{f}_a) repelling agent a . To determine the magnitude of the repulsion vector for each individual obstacle, force functions $U(|\vec{d}_{ao}|)$ for static obstacles and $V(|\vec{d}_{ab}|)$ are used. $|\vec{d}_{ao}|$ or $|\vec{d}_{ab}|$ is the distance between agents a and obstacle o or agent b . $U(|\vec{d}_{ao}|)$ is implemented as follows: The distance vector $\vec{d}_{ao} = \vec{o} - \vec{a}$, r_a is the radius of a , and m_a is the Goffman ellipse distance of agent a . r_a ensures the distance to the border of agent a is the effective distance and m_a increases the repulsive force on agents.

$$U(|\vec{d}_{ao}|) = \frac{m_a + r_a - |\vec{d}_{ao}|}{(|\vec{d}_{ao}| - r_a)^2} \tag{1}$$

A selection process was added based on Goffman’s research on how pedestrians observe other pedestrians. In Fig. 1 a, a visual representation is given to illustrate which agents exert influence, with the grey agents being excluded. $V(|\vec{d}_{ab}|)$ is implemented such that, as stated by Goffman, the radial orientation determines how influential another pedestrian is, i.e., a pedestrian to the side exerts less social pressure than in front of the agent considered. Let $L(a, b)$ be the distance between a and the ellipse contour at angle \hat{d}_{ab} , and let $L(a, min)$ be the distance between a and the ellipse contour on its minor axis. Then, the scalar function $V(|\vec{d}_{ab}|)$ is given by

$$V(|\vec{d}_{ab}|) = |\vec{d}_{ab}|^{-2} \cdot \frac{L(a, b)}{L(a, min)} \tag{2}$$

4 Experiments

The corridor and intersection experiments aim at producing environments that represent optimized geometric setups. Each setup contains agent starting areas and a collection of static obstacles \mathcal{O} . Each starting area has a corresponding goal area. Agents are spawned in their starting areas and will attempt to find a route to their goal. When an agent reaches its goal area, it is removed from the simulation. In this context, maximizing pedestrian efficiency means minimizing the average time between agents being spawned and removed. Efficiency is measured in percentages where 100 % equals the mean time agents need to complete their route if there are no static or dynamic obstacles.

To test whether an evolutionary strategy can find a better environment than a human designer, we have provided three basic environments for both the corridor and intersection experiment. For both optimization experiments we have used a so-called $(1, \lambda)$ -ES as described by Bäck [1], with $\lambda = 10$.

4.1 Simulator Validation

For the different types of emergent behaviour described in Sect. 2.2, environments were created that should show the emergent behavior if used by real pedestrians. The results are interpreted by looking at a real time representation of a pedestrian simulation and noting the similarities and dissimilarities between the pedestrian and agent behaviour.

4.2 Corridor and Intersection

The corridor setup shown in Fig. 3a (left) has agents starting in areas Z and X with the goal of the agents spawning in area Z to reach area X , and vice versa. Alongside the corridor in between areas Z and X are five static obstacles a, b, c, d, e . These obstacles are square pillars that can be moved along the width of the corridor (denoted by the dotted line) and changed in radius r . Each obstacle is represented for the evolutionary strategy by two real valued dimensions, one representing the position along the width of the corridor and the second one representing the obstacle's radius.

The intersection setup in Fig. 3b (right) consists of two corridors intersecting in a 90° angle with the overlapping, traversable part of the two corridors being area K . The starting areas where the agents spawn are W, X, Y and Z . The goal area for the agents that spawn in W is Y , for Y it is W , for X it is Z and for Z it is X .

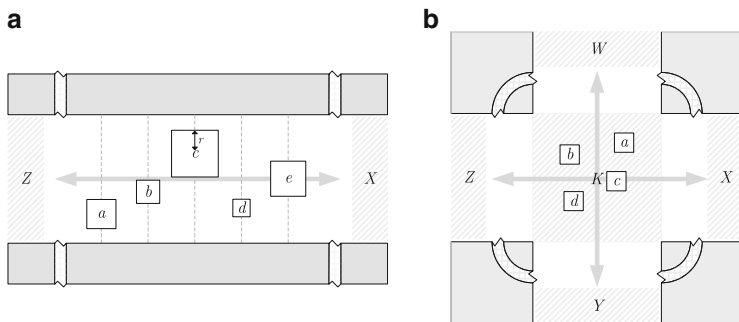


Fig. 3 Range of obstacle movement in experiments. (a) Corridor setup. (b) Intersection setup

In area K , four square static obstacles a, b, c, d of identical size are placed. They can be moved to any place within K while not overlapping the border of K and not overlapping any of the other obstacles. Each obstacle is represented by two real valued dimensions, being the x and y position of the obstacle in the intersection.

5 Results

5.1 Simulator Validation

Overall, the expected emergent behaviour is observed, although some more profound than others. The clogging effect is very much present in the simulator. Lane formation also occurs readily as is seen with pedestrians. The number of agents that make up a lane also concurs with that observed amongst pedestrians. Finally, the time one particular lane survives is also, as with pedestrians, very limited.

5.2 Environment Optimization

Three runs were executed for both the intersection and corridor experiment. The stop condition for optimization runs is based on an internal convergence measure of the solution set maintained by the optimizer. In terms of computational effort, the average completion time per optimization run is about 1–2 days. In the corridor runs convergence of optimization was achieved, although this seemed to be lacking in the intersection runs. An illustration on how these runs develop can be seen in Fig. 4 for the corridor (left) and intersection (right). Note that the fitness values used by the

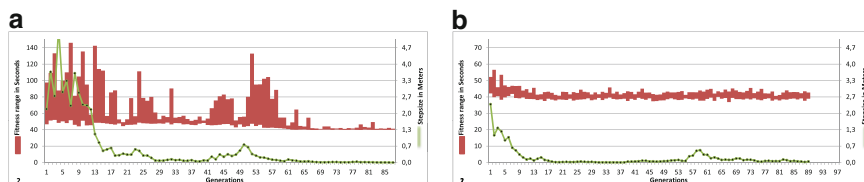


Fig. 4 Progression of the evolutionary strategy over time, measured in generations of the optimizer. Pedestrian performance range is from the child with the best fitness to the child with the worst fitness of that generation. (a) ES C2. (b) ES I2

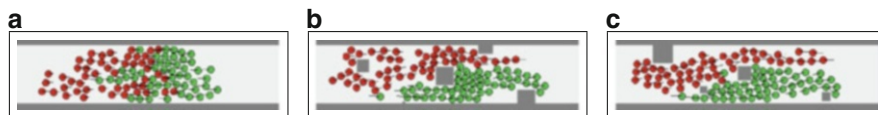


Fig. 5 Visual representation of the empty corridor environment (*left*) and two optimized corridor environments during a fitness evaluation. (a) Empty. (b) ES C1. (c) ES C2

optimizer are the average number of calculation steps needed for an agent between spawning and reaching its goal. The best solution found in each run, i.e., the solution with the best fitness, is used as a representative. They are designated with a C for corridor and I for intersection runs, e.g., the best solution from the second corridor run is labeled C2.

For each environment and spawn frequency,¹ 50 simulations are performed, and the average performance is measured. The results are presented as a percentage of the agent potential. An agent's potential is the time taken from start to goal if no obstacles are present. In the corridor setup this is 32.35 s, and for the intersection setup this is 19.08 s. All results are summarized in Fig. 6 showing how the fixed and the optimized environment's geometry efficiency changes with increasing spawn frequency. For large spawn frequency, optimization can clearly increase efficiency on the corridor.

A visual representation of the two optimized environments with the best fitness in the corridor experiment and the typical emerging agent behavior is shown in Fig. 5b, c, in comparison to the empty corridor in (a).

For the evolution strategy runs for the corridor experiment, both stepsizes and fitness ranges decrease over time. This indicates that convergence is taking place. Moreover, at high spawn frequency all optimized solutions show a better mean pedestrian efficiency than an empty corridor. In contrast, the runs for the intersection experiment do not exhibit the same convergence. In the intersection runs the stepsize decreases over time, but the solutions found do not improve significantly. Also, the

¹Spawn frequency, in Hz, refers to how often agents are added to a starting area.

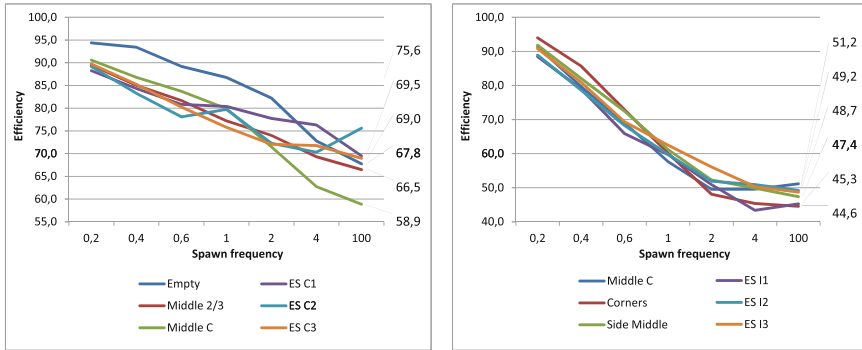


Fig. 6 Mean fitness comparison of the six corridor (left) and intersection (right) environments

fitness range within one generation, although fluctuating, remains small and does not decrease over time. In case of the intersection, no improved configuration (when compared to the empty intersection) could be found.

6 Conclusions

The simulator has been shown to produce realistic behaviour in environments where circular flows do not occur. Although the stochastic simulation generates a noisy landscape for the optimization, at high density, solutions can be discriminated and convergence takes place.

In the corridor experiment it was found that all optimization runs produced a better solution at high density than the empty corridor (best “hand-designed” solution). This is the most promising result of this research as it not only shows that placing obstacles in pedestrian areas can lead to an increased pedestrian efficiency but also suggests that the location and size of obstacles can be found by an optimizer.

References

1. Th. Bäck. *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, Oxford, UK, 1996.
2. C. Burstedde, K. Klauck, A. Schadschneider, and J. Zittartz. Simulation of pedestrian dynamics using a two-dimensional cellular automaton. *Physica A: Statistical Mechanics and its Applications*, 295(3–4):507–525, 2001.
3. P. G. Gipps and B. Marksjo. *Mathematics and Computers in Simulation*, 27:95–105, 1985.
4. E. Goffman. *Relations In Public: Microstudies In The Public Order*. Basic Books, 1971.
5. D Helbing, A Johansson, and H. Z. Al-Abideen. The dynamics of crowd disasters: An empirical study. *Phys. Rev. E*, 75:046109, 2007.

6. D. Helbing and P. Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, 1995.
7. A. Kirchner and A. Schadschneider. Simulation of evacuation processes using a bionics-inspired cellular automaton model for pedestrian dynamics. *Physica A*, 312(1–2):260–276, 2002.
8. S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at <http://planning.cs.uiuc.edu/>.
9. S. Okazaki. *Trans. of A.I.J.*, 283:111–119, 1979.