CHAPTER SEVEN

Microscopic models and simulation

Dorine C. Duives[*]
Department of Transport & Planning, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft, The Netherlands
*Corresponding author. e-mail address: d.c.duives@tudelft.nl

Contents

Abstract

The movement patterns of pedestrian crowds at train stations, large-scale events, and cities are intricate. Due to the complex interplay between pedestrians, the physical environment, the physiological environment, and the available information, predicting how a pedestrian crowd will move is challenging. Microscopic pedestrian simulation models can help identify which movement patterns will arise, when and where, and to what extent the predicted movement patterns might potentially be problematic. Various pedestrian simulation models have been introduced to model pedestrians' operational movement dynamics. This chapter explores only one of these simulation model categories: microscopic pedestrian simulation models. We introduce a wide variety of microscopic model types currently used to model pedestrian movement dynamics at a microscopic level, including the Cellular Automata model (CA), the Social Force model

Advances in Transport Policy and Planning, Volume 15
ISSN 2543-0009, https://doi.org/10.1016/bs.atpp.2025.03.004

(SF), the predictive collision avoidance model (PCA), the Optimal Step Model (OSM), the Discrete Choice model (DC), and the data-driven artificial neural network model (ANN). The overall movement dynamics of each agent result from the interactions between agents, their physical and physiological environment, and available information. At the same time, each model type simulates the movement dynamics of agents in a particular way, resulting in slightly different operational and global movement dynamics. This chapter presents the most naive version of each model type, as this best describes the agents' general choice behavior and movement dynamics. In addition, some interesting extensions, benefits, and challenges of each model type are discussed.

1. Modelling pedestrian dynamics microscopically

The movement patterns of pedestrian crowds at train stations, large-scale events, and cities are intricate. Due to the complex interplay between pedestrians, the physical environment, the physiological environment, and the context, predicting how a pedestrian crowd will move is challenging. Microscopic pedestrian simulation models can help identify which movement patterns will arise when and where and to what extent the predicted movement patterns might potentially be problematic.

A pedestrian on the move makes six choices when traveling from an origin to a destination. Fig. 1 provides an overview of these choices and the interplay between the choices. While these choices are not strictly hierarchical, there is a particular order due to the timescales at which a pedestrian must make the travel choices.

One needs to model all six choice levels to model pedestrian movements comprehensively. Yet, most pedestrian simulation models predominantly simulate the lowest travel choice level. That is, pedestrian simulation models are used to dominantly simulate the operational movement choices of pedestrians within a piece of infrastructure, where the other five choices (i.e., activity, destination, departure time, and route choice) form a (static) input to the model. Mode choice is irrelevant in the context of pedestrian simulation, as only walking individuals are part of a pedestrian simulation model.

Various pedestrian simulation models have been introduced to model pedestrians' operational movement dynamics. Roughly, literature distinguishes three big model categories. Macroscopic pedestrian models, such as the continuum models of (Hoogendoorn et al., 2014) and (Hughes, 2002), see pedestrian crowds as moving fluids or gasses. Microscopic pedestrian models, such as the Social force model (Helbing and Molnar, 1995) and the Cellular Automata (Blue and Adler, 1998), model the actions of each pedestrian and simulate how the combination of all these individual actions leads to aggregate

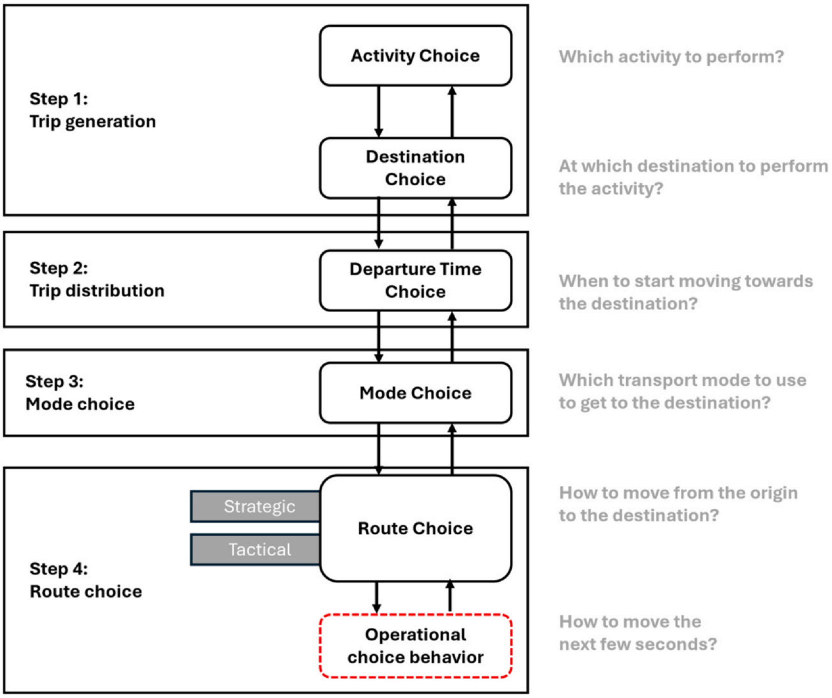

Fig. 1 Travel choice level framework, which combines the four-step model and the choice level framework of (Hoogendoorn and Bovy, 2004).

crowd dynamics. Mesoscopic pedestrian models, such as SimPed (Daamen, 2002), model each agent's actions using aggregate flow properties.

This chapter explores only one of these three four simulation model categories, namely microscopic pedestrian simulation models. We introduce a wide variety of microscopic model types currently used to model pedestrian movement dynamics at a microscopic level. Each of these describes pedestrian movement dynamics slightly differently. Due to differences in the model formulation, the resulting microscopic model types also predict slightly different pedestrian movement dynamics. Consequently, basic knowledge of the model's mathematical formulation is required to understand the fundamental differences between the model types and how the mathematical formulation of the behavior of individual agents influences the crowd's movement dynamics that a simulation model predicts.

In the remainder of this chapter, we first detail the different microscopic pedestrian model types in Section 2. Accordingly, Section 3 provides an overview of the capabilities of each model type. To do so, we adopt a part of the framework presented in Duives et al. (2013).

2. Microscopic model types

The literature on microscopic pedestrian simulation models up to date can be subdivided into model types with fundamentally different ways of capturing the physical movement process in a set of mathematical equations. Some model the world as a continuous space through which balls find the path of least resistance, while others approach it more like a game of chess, where pieces move on a predefined grid.

In total, six fundamentally different model types can be distinguished, namely the Rule-based models (RB), the Social Force model (SF), the predictive collision avoidance model (PCA), the Optimal Step Model (OSM), the Discrete Choice model (DC), and the data-driven artificial neural network model (ANN). All these microscopic model types have in common that each pedestrian's actions are explicitly programmed. The overall movement dynamics result from the interactions between agents, their physical and physiological environment, and available information. At the same time, each model type simulates the movement dynamics of agents in a particular way, resulting in slightly different operational and global movement dynamics.

This chapter presents the most naive version of each model type, as this best describes the agents' general choice behavior and movement dynamics. Each section also mentions some interesting extensions of each model type to provide ideas on enhancing its capabilities. Lastly, each model type's main benefits and challenges are identified.

From here onward, we will use the word 'agent' to describe an object resembling a pedestrian that a simulation model moves through space according to a set of predefined rules. The word "pedestrian" is reserved for real individuals.

2.1 Rule-based models

The rule-based model is the first category of models ever used to model active mode movement behavior. Conceptually, this model type is pretty straightforward. The agents' movement (and choice) behavior results from a set of 'simple' rules. Depending on the agent's characteristics, this simple set of generic rules can still result in heterogeneous movement and choice behavior. For example, speed differences between agents can be implemented in a rule-based model by increasing or decreasing the probability that an agent will make a step during a timestep.

When rule-based models are applied to simulate pedestrian movement dynamics, two main model types are adopted: (1) a Cellular Automata (CA) or (2) an Agent-based model (ABM). While the first model is predominantly used to directly model the agents' movement dynamics without interference of thought processes, the second model is often adopted to simulate the impact of the choice behavior of agents at a strategic and tactical level on the movement dynamics of agents (e.g., Blue and Adler, 1998; Irnich et al., 2022).

2.1.1 Mathematical formulation CA model

The Cellular Automata (CA) was first introduced by (Blue and Adler, 1998). After its introduction, CA models quickly grew to become one of the leading model types used to simulate crowd movement dynamics. The CA model discretizes space as well as time. This model allows agents to move through space by making them move like chess pieces over a rectangular (or other equidistant-shaped lattice) grid on the floor (Fig. 2). This grid, also coined a floor field, represents the memory contained within the model. Depending on the current information contained within the grid, agents determine their next step. Whether an agent moves to another cell and which exact cell is visited next is the result of an optimization of movement opportunities or a probabilistic process.

Consequently, the floor field determines, to a large extent, the local choice behavior of the agents. All information available within the infrastructure is quantified in one or more floor fields to allow the floor field to do so. The most naive implementation of CA models contains only three pieces of information in each grid cell of the floor field, namely (1) the distance from the cell to the nearest exit, (2) whether an agent can visit the cell, and (3) whether another agent currently occupies the cell. More complex CAs also include information regarding the movements of other agents, obstacles (e.g., distance to walls, the function of the cell, the cell's physiological status), and guidance information (e.g., indicated routing) in the infrastructure.

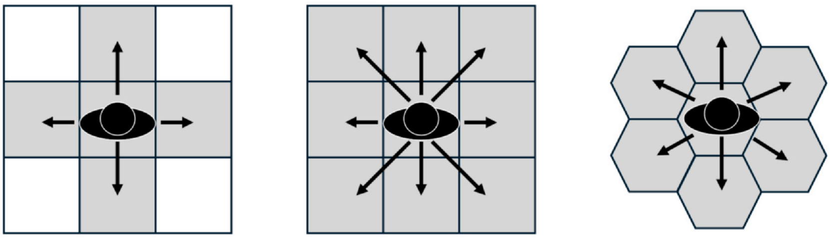

Fig. 2 Popular neighborhoods for a CA model, left: Von Neumann neighborhood on a rectangular grid, middle: Moore neighborhood on a rectangular grid, right: Von Neumann & Moore neighborhood on a hexagonal grid.

2.1.2 Updating scheme CA model

If many agents are simultaneously present within space, two agents may want to enter the same cell simultaneously. How the competition for space by agents is solved depends on the updating strategy. In general, two update strategies are used for CAs, namely sequential and parallel updating strategies. The more simple sequential update strategy randomly selects an agent from the list of agents that need to be moved, determines its next option given the transition probability, the current position of all other agents, and obstacles, and moves this agent (see Algorithm 1). Note that the algorithm is the authors' interpretation of the descriptions of CA models found in the literature. At each timestep, all agents are moved one by one in a random order. As such, the 'conflicts' are handled using a 'first-come-first-serve' strategy, where the agents' probability of being selected determines their movement opportunities. The algorithm attempts to move each agent only once per timestep. Thus, the required computational effort for a sequential update scheme is limited.

Algorithm 1. Example code of CA with sequential updating.

```
procedure SequentialUpdatePosition(A, P)
    for a ∈ A do                                              ▷ for all agents
        Initialize (c ∈ C_candidate)                          ▷ list of candidate cells
        Initialize c_{a,new} = ∅                              ▷ new position agent a
        while c_{a,new} = ∅ and C_candidate ≠ ∅ do
            c ~ Uniform(C_candidate)
            P_step ← P(c)
            P_draw ~ U(0,1)
            if P_draw < P_step then
                c_{a,new} ← c                                 ▷ agent a moves to cell c
            else
                C_candidate ← C_candidate \ {c}               ▷ remove c from list
            end if
        end while
    end for

    return P

end procedure
```

The other update strategy, (stochastic) parallel update, determines the best movement options for all agents simultaneously and only moves agents when no conflicts exist. One potential update schemes is presented in Algorithm 2. The algorithm will first determine which agents can step without any conflicts. Accordingly, it determines a movement option for all remaining agents, given the new information on the new positioning of the already moved agents.

This loop continues until all conflicts are resolved or no better solution can be found. When a sequential update scheme is left unbounded, this process can require many iterations when large crowds are simulated. In practice, an upper limit is often set on the number of iterations within a timestep.

Algorithm 2. Parallel update scheme for agents in CA.

```
procedure ParallelUpdatePosition(A, P)
    Initialize C_new = ∅                                  ▷ Set of new cell positions
    Initialize A_move = A                                 ▷ Set of agents who need to move
    while A_move ≠ ∅ do
        for a ∈ A_move do
            Initialize C_candidate                        ▷ List of candidate cells
            while C_new(a) = ∅  and  C_candidate ≠ ∅ do
                c_can ~ Uniform(C_candidate)              ▷ Random select candidate
                P_step ← P(c_can)
                P_draw ~ U(0, 1)
                if P_draw < P_step then
                    C_new(a) ← c_can                      ▷ Move a to new cell
                    break                                 ▷ Go to next agent
                else
                    C_candidate ← C_candidate \ {c_can}   ▷ Remove c_can from list
                end if
            end while
        end for
        for each a, i ∈ A_move do                         ▷ Check agents in same cell
            if ∀a∀i  (a ≠ i) and (C_new(a) = C_new(i)) then
                C_new(a) = ∅                              ▷ Erase candidate solution
                A_move ← A_move + [a]                     ▷ Add agent a to the list A_move
                break
            end if
        end for
    end while
    P ← C_new
    return P
end procedure
```

2.1.3 Recent CA model adaptations

The naive uni-directional CA model of (Blue and Adler, 1998) has been extended by various researchers in the last three decades. In the beginning, especially more complex update rules to deal with bidirectional traffic (e.g., (Blue and Adler, 1999)), four-directional traffic (e.g., (Blue and Adler, 2000)) and collective phenomena (e.g., Schadschneider et al. (2002)). Later, more complex behavioral rules were introduced to deal with crowd dynamics (e.g., (Bazior et al., 2020)). Among other things, researchers developed models that can simulate grouping (e.g., Bandini et al., 2011; Zhang et al., 2023b), corner-turning (Dias and Lovreglio, 2018), stair walking (Chen and Lu, 2024; Xie et al., 2022), queuing (Hu et al., 2018) and the impact of falling (Ma et al., 2024) in more complex environments. More and more, the CA model has changed from a model in its own right to the operational walking model of an Agent-Based Model (ABM), where complex decision-making processes are translated into movement probabilities. Risk perception, herding, and leader-follower behavior are just a few of the behaviors for which CA-based applications exist (e.g., Huo et al., 2024; Hu et al., 2015; Irnich et al., 2022).

2.1.4 Benefits and challenges CA models

The main benefit of a CA model is its high computational efficiency and the fact that the walking behavior results from a fairly simple set of rules. As such, the model can easily simulate very large crowds while the movement solutions remain easily tractable. An often recorded downside of a CA model includes the challenges of designing a nicely fitting floor field for real-life cases, which generally are not built to follow a systematic grid of equally dimensioned cells. Moreover, the often adopted rectangular floor field might result in unrealistic (discrete) movement behavior, such as too low capacity and abrupt changes in direction and speed. Thirdly, friction between agents cannot be simulated as agents either occupy or cannot occupy a grid cell. Lastly, researchers indicate that it is challenging to incorporate less tangible features of the environment in the simulation, for example, light conditions and lane preferences.

2.2 Social force models

This second category of models was introduced approximately at the same time as the CA model by (Helbing and Molnar, 1995). In contrast to the CA, the Social Force (SF) model allows agents to move continuously through space. Agents and obstacles are modeled as rigid bodies that are influenced by attractive and repulsive forces created by surrounding objects and agents. Moreover, agents do not choose between a set of movement options but compute the most optimal movement option given their current surroundings.

This option is the result of the balancing of several forces. Social force models assume that each object (e.g., walls, agents, guidance signs, storefronts, exits) in the environment emits an attractive or repulsive force. Please note that objects do not necessarily have a physical representation in the walking space of the agent (e.g., sound systems and signs).

The closer an agent is to an object, the larger the force emitted by the object. Consequently, the acceleration of an agent is calculated as the sum of the forces divided by the mass of the agent. See equation (Eq. (1)) where F represents the social force, m_i the mass of agent i, $\vec{a}_i$ the acceleration of agent i. The new speed ($\vec{v}_i(t)$) and location ($\vec{x}_i(t)$) of an agent are accordingly computed by integrating the acceleration over timestep δt, see Eq. (2) and (3).

$$\vec{a}_i = \frac{\sum F}{m_i} \tag{1}$$

$$\vec{v}_i(t + \delta t) = \vec{v}_i(t) + \vec{a}_i * \delta t \tag{2}$$

$$\vec{x}_i(t + \delta t) = \vec{x}_i(t) + \vec{v}_i * \delta t \tag{3}$$

2.2.1 Mathematical formulation naive SF model

Several force-based models have been presented. The most well-known model is the naive Social Force model (SF) by (Helbing and Molnar, 1995). This model is specified by Eqs. (4)–(7). In these equations, a_p^0 represents the preferred acceleration, v_p^0 the preferred speed of agent i, τ the reaction time, $r_{p,w}$ and $r_{i,j}$ the shortest distance from agent p to the wall w and other pedestrians q, $r_{w,max}$ and $r_{p,max}$ the maximum distance at which walls and other pedestrians are accounted for C_{wall} and C_{agents} constants that determine the maximum strength of the forces emitted by walls and other pedestrians.

$$\vec{a}_p(t) = \vec{a}_p^0 + \frac{\sum_w \vec{F}_w + \sum_p \vec{F}_{p,q}}{m_p} \tag{4}$$

$$\vec{a}_p^0 = \frac{1}{\tau}(v_p^0 * \vec{e}_p - \vec{v}_p) \tag{5}$$

$$\vec{F}_{p,w} = -1 * \vec{e}_{p,w} * C_{wall} * e^{-\frac{\| r_{p,w} \|}{r_{w,max}}} \tag{6}$$

$$\vec{F}_{p,q} = -1 * \vec{e}_{p,a} * C_{agents} * e^{-\frac{\| r_{p,q} \|}{r_{p,max}}} \tag{7}$$

2.2.2 Updating scheme SF model

In contrast to the CA, only parallel updating is used in force-based models because the update steps are so small that the difference between parallel and sequential update schemes is negligible. Algorithm 3 depicts the update procedure. All agents are randomly visited once per timestep and will all move during each timestep.

Algorithm 3. Social force update scheme.

```
procedure SFUpdate(A, V, X)

    Initialize A_new = A                          ▷ Set of agents to be moved

    while A_new ≠ ∅ do
        a ~ Uniform(A_new                         ▷ Random select agent
        Compute acceleration towards goal a⃗_p^0(t)
        Compute wall forces F_{p,w}(t)
        Compute agent forces F_{p,q}(t)
        Compute new acceleration a⃗_p(t)
        Compute new velocity v⃗_p(t)
        Compute new position x⃗_p(t)
        X(a) ← x⃗_p(t)                            ▷ Record new location
        V(a) ← v⃗_p(t)                            ▷ Record new location
        A_new ← A_new \ {a}                       ▷ Remove agent a from A_new set
    end while

    return V, X

end procedure
```

2.2.3 Recent SF model adaptations

After its conception, the naive SF model quickly became one of the most popular microscopic pedestrian simulation models. Over time, many improvements have been proposed to reduce the most pressing artifacts of the model, and extensions of the model have been proposed to deal with unique scenarios or pieces of infrastructure (Chen et al., 2018). For instance, the naive SF model could not always realistically resolve the bi-directional collision avoidance. Various researchers have proposed fixes to improve the collision avoidance behavior of the SF (e.g., Lee et al., 2016; Heliövaara et al., 2012). Some extensions of the agents' perception of other objects have also been introduced to be able to model anisotropy (e.g., Cristiani et al., 2011; Garcia et al., 2023). In addition, several attempts were undertaken to extend the capabilities of the SF w.r.t. the simulation of social interactions in crowds. (Moussaid et al., 2010) and (Xu and Duh, 2009), for instance, presented two strategies to incorporate grouping behavior in the naive SF. Other researchers also incorporated additional force-based elements to capture overtaking, stopping, and waiting behaviors (e.g., Yuen and Lee, 2012; Seyfried et al., 2006; Johansson et al., 2015), and (Parisi et al., 2009)). (Yu and Johansson, 2007) mode changes to the formulation of the agent-to-agent interaction force to model irregular displacement of pedestrians during turbulence in high-density crowds. (Zheng et al., 2020) developed an extension of the SF model that changes the dynamics of the individual based on their emotional state. Lastly, some researchers have developed extensions of the naive SF for the SF to be able to realistically cope with special types of infrastructure, such as stairs and corners (among others, (Dias et al., 2019; Qu et al., 2014)).

2.2.4 Benefits and challenges SF models

The SF model is known to simulate the behavior of pedestrians quite realistically if the model is correctly calibrated and validated for a specific scenario. More importantly, the resulting interaction behavior between agents has been face-validated on many occasions. Moreover, the use of force allows agents to move in the same space under extreme circumstances (e.g., extremely large crowds, high pressure). Consequently, friction can be modeled using a force-based model. Lastly, as long as the interaction behavior between an agent and an entity can be translated into a force, it is possible to incorporate features of the environment (agents, objects, and guidance information) into the simulation model.

However, because the model is reasonably complex and agents orient themselves in relation to surrounding obstacles, the computational effort of this model is relatively high. The procedures for computing the forces of walls and other pedestrians, which change at every timestep, are costly. In addition, because the SF model is flexible with respect to the type of interactions that can be modeled, the parameter set is a sophisticated SF, often extensive. This makes the calibration and validation of the SF model cumbersome.

2.3 Predictive collision avoidance models

A third, more recent addition to the microscopic pedestrian simulation model landscape is the predictive collision avoidance (PCA) model, also called the reactive model, velocity-based model or collision-free velocity model. To the authors' knowledge, this model type was first introduced by Paris et al. (2007). This type of model assumes that pedestrians optimize the usage of the available space and attempt to avoid collisions as much as possible.

Predictive collision avoidance models predict the trajectory of all surrounding agents. They establish a set of admissible velocities that will not lead to a collision. Within this set, the model chooses the velocity closest to their preferred walking velocity, thereby limiting their acceleration and deceleration. How these models determine this set of admissible velocities depends on the specific mathematical operationalization of each model. The reactive model by (Paris et al., 2007) establishes the lower and upper boundaries of the speed and direction that will not lead to a collision. The Reciprocal Velocity Obstacle model of (Van Den Berg et al., 2008) starts with a full range of velocities and iteratively deletes all potential velocities that lead to a collision from the set, thus leaving the set of acceptable velocities. The Linear Trajectory Avoidance (LTA) model by (Pellegrini et al., 2009) determines the cost of each potential velocity and accordingly adopts the velocity with the lowest cost. The PCA model by (Karamouzas and Overmars, 2010) follows a similar approach.

2.3.1 Mathematical formulation of the PCA model

This section presents the model of (Karamouzas and Overmars, 2010), which provides a very efficient algorithm and a very intuitive mathematical description. This particular PCA model approaches collision avoidance as an optimization problem. This PCA model gives every agent the optimal velocity (speed and direction) that allows the agent to evade all obstacles and other agents while moving towards their goal as efficiently as possible.

To do so, it first determines the set of available orientations O_a and speeds V_a for every agent a and combines the two sets into a set of Feasible Avoidance Velocities (FAV_a) (see Eq. (8)–(10). Here, θ_{des} represents the desired direction of agent a, θ_{amax} is the maximum deviation from the desired direction that can be achieved within one timestep. Similarly, $v_{p,pref}$ represents the preferred speed of pedestrian p, $v_{p,max}$ the maximum change in speed that pedestrian p can achieve by agent a in timestep t. tc_{min} represents the time to the collision, i.e., the time at which collisions between agent a and surrounding agents are imminent. tc_{max} is the maximum time to the collision at which a pedestrian still anticipates the movements of another agent.

$$O_a = \{n_\theta | \theta \in [\theta_{a,des} - \theta_{a,max}, \theta_{a,des} + \theta_{a,max}]\} \tag{8}$$

$$V_a = \begin{cases} v | v \in [0, v_{max}] & \text{if } 0 \leq tc \leq tc_{min}, \\ v | v \in [v_{a,pref}, v_{a,pref} \pm v_{max}] & \text{if } tc_{min} \leq tc \leq tc_{max}, \\ v_{pref} & \text{if } tc_{max} \leq tc \end{cases} \tag{9}$$

$$FAV_a = \{(v, n_\theta) | v \in V_a \wedge n_\theta \in O_a\} \tag{10}$$

Afterward, the total energy expenditure of each pair in the set FAV_a is computed using Eq. (11). Here, the first two elements describe the kinetic energy consumption resulting from a directional ($\Delta\phi$) and speed ($\|\vec{v}_{can}\| - v$) change. The third element describes the kinetic energy consumption resulting from a change in direction ($\|\vec{v}_{can} - \vec{v}_{p,des}\|$) required to move toward its desired destination. The last element describes the potential costs of a collision, where tc describes the minimum predicted collision time between agent p and the set of agents P. Here, α, β, γ, and δ are constants that together determine the relative influence of the four energy types.

$$\begin{aligned} E(v, n_\theta) = {} & \alpha\left(1 - \frac{\cos \Delta\phi}{2}\right) + \beta\frac{\|\vec{v}_{can}\| - v}{v_{max}} \\ & + \gamma\frac{\|\vec{v}_{can} - \vec{v}_{p,des}\|}{2 * v_{max}} + \delta\frac{tc_{max} - tc}{tc_{max}} \end{aligned} \tag{11}$$

The optimal velocity for agent p in a given timestep t is the option that minimizes the total additional required energy (see Eq. (12)).

$$\vec{v}_p(t + \delta t) = \arg \min_{\vec{v}_{can} \in FAV_p} E(v, n_\theta) \tag{12}$$

$$\vec{x}_p(t + \delta t) = \vec{x}_p(t) + \vec{v}_p * \delta t \tag{13}$$

2.3.2 Update scheme PCA model

The complete algorithm is depicted in Algorithm 4. Each agent is only visited by the algorithm once per timestep. Consequently, a sequential update scheme is often applied to speed up the simulations.

Algorithm 4. Predictive collision avoidance update scheme.

---

procedure PCAUPDATE(A, V, X)

    `Initialize` $A_{new} = A$ ▷ Set of agents to be moved

    while $A_{new} \neq \varnothing$ **do**

        $a \sim$ `Uniform(`A_{new} ▷ Random select agent

        Determine current location $\vec{x}_a(t)$

        Determine current velocity $\vec{v}_a(t)$

        Initialize set of admissible orientations O_a

        Initialize set of admissible speeds V_a

        Determine the total energy of solutions $E(v, n_{theta})$

        Select the optimal solution

        Determine the new velocity $\vec{v}_a(t + \delta t)$

        Determine the new location $\vec{x}_a(t + \delta t)$

        $X(a) \leftarrow \vec{x}_a(t + \delta t)$ ▷ Record new location

        $V(a) \leftarrow \vec{v}_a(t + \delta t)$ ▷ Record new velocity

        $A_{new} \leftarrow A_{new} \setminus \{a\}$ ▷ Remove agent a from A_{new} set

    end while

    return V, X

end procedure

In practice, an infinite set of feasible velocities exists. Therefore, the domains of O and V are often restricted to a discrete set of samples to improve the computational efficiency of PCA models. Here, the modeler has to balance the need for an efficient model with the potential introduction of computational artifacts due to suboptimal choices. In general, more granular discretization allows for smoother pedestrian dynamics. At the same time, more granular discretization also rapidly decreases the computational efficiency of the model.

2.3.3 Recent PCA model adaptations

Very few adaptations have been proposed for the naive PCA models. To the author's knowledge, only (Guy et al., 2009) proposed a method to parallelize the PCA model for simulations with large crowds (>100, 000 agents). Later, various adaptations were made to enhance collision avoidance, group movements, and dynamics under high densities. For instance, (Xu et al., 2021) additional elements are introduced in the predictive collision avoidance model to improve the anticipation of agents. (Ren et al., 2017) introduced the Velocity Connection; a set of velocities that keep agents moving together while moving through the infrastructure. To improve the simulation of self-organization behavior, (Zhang et al., 2023a) introduces a competition coefficient that describes agents' direction preference in spaces with less competition for the same space. In addition, (Kim et al., 2015) introduced external forces in the PCA model to model physical interactions in high-density crowds. All of these show that the basic structure of the naive PCA models is very flexible and can be extended to simulate more specialist behaviors.

2.3.4 Benefits and challenges PCA models

Predictive Collision Avoidance models compute the optimal collision-free velocity for each agent at each time step. Consequently, in confined and crowded environments, the agent may be unable to find an optimal solution due to the discretization of the choice set. The authors experienced that PCA models produce simulation artifacts at very high densities. For instance, the model adjusts the agents' direction or speed very quickly, or agents stop very abruptly. Most PCA models also show erratic behavior when densities increase rapidly.

At the same time, the model's underlying mathematical structure is very flexible. As long as one can describe a factor's impact on energy consumption, one can incorporate the factor into the model. This allows for relatively simple extensions of the modeling structure in accordance with the latest insights into pedestrian interaction behavior. In addition, the physics-oriented approach also allows one to adapt this model type to other modes of micro-mobility, such as cyclists, e-scooters, etc.

2.4 Optimal step model

The Optimal Steps Model (OSM) is loosely inspired by the CA. The OSM is a non-differential microscopic model that adopts the idea that positive and negative potentials impact the next step in an agent's direct neighborhood. In contrast to other microscopic modeling approaches, agents make discrete steps to move forward. "They do not glide along smooth trajectories that resemble imaginary rails as in force-based models or hop from cell to cell as in cellular automata." Köster et al. (2016). An agent's speed adaptations result from interactions with individuals in the direct neighborhood.

2.4.1 Mathematical description of the OSM

To move, any agent in the OSM uses the dynamic potential field to determine the optimal next step. The total potential $P_{total}(p, x, t)$ of any pedestrian p a point in the space $x \in \mathbb{R}^2$ at time t is determined using Eq. (14).

$$P_{total}(a, x, t) = P_{attract}(x, x_a, t) + \sum_{j=1}^{m} P_{obstacle}(j, x, t) + \sum_{q=1, a \neq q}^{n} P_{ped}(q, x, t) \quad (14)$$

Here, $P_{attract}(x, x_a, t)$ represents the attractive potential of the target at time t at location x given the current location of the agent of focus which is located at x_a, $P_{obstacle}(j, x, t)$ the repulsive potential of object j at location x, and $P_{ped}(q, x, t)$ the repulsive potential of pedestrian q at location x.

Within the OSM, the potential strength depends on the distance between location x and the element that causes the potential. For $P_{obstacle}(j, x, t)$ the obstacle potential is defined as

$$P_{obstacle}(j, x_o, t) = \begin{cases} \mu_o & \text{if } d_o(j, x, t) \leq g_a/2, \\ \nu_o \times \exp\left[-\alpha_o \times d_o(j, x_o, t)^{b_o}\right] & \text{if } g_a/2 < d_o(j, x_o, t) \leq h_o, \\ 0 & \text{else} \end{cases} \quad (15)$$

Here, $d_o(j, x, t)$ represents the distance between the obstacle and point x, g_a is the radius of the torso of a body, and h_o the maximum distance at which an obstacle influences an agent. δ_1 and δ_2 are exponential decay functions representing the potential strength as a function of the distance between x and obstacle j. μ_o, ν_o, α_o, b_0, h_0 are model parameters that can be calibrated. To ensure agents can move through bottlenecks, the obstacle potential is a weak but far-reaching potential. The exact definition of the potential functions and the suggested parameter settings can be found in (Seitz and Köster, 2012).

The potential strength of the repulsion due to neighboring agents is determined by:

$$P_{ped}(a, x, t) = \begin{cases} \mu_a & \text{if } d_a(q, x, t) \leq g_a, \\ \nu_a \times \exp\left[-\alpha_a \times d_a(q, x, t)^{b_a}\right] & \text{if } g_a < d_a(q, x, t) \leq g_a + h_a, \\ 0 & \text{else} \end{cases} \quad (16)$$

Here, μ_a is the maximum potential an agent experiences when the body diameters entirely overlap with another agent, which is set to a very high value (e.g., 1000) to ensure the probability of overlap is limited. ν_a, α_a, b_a, and h_a are model parameters that can be calibrated depending on the context and flow scenario. Here, the overlap between body circumferences can be used to estimate compression of the torso due to pressure or friction.

Accordingly, the new position of each agent is found by solving a two-dimensional optimization problem with one equality constraint (see (17)). Here, $d_a(x)$ represents the Euclidian distance between the current position $x_a(t)$ of agent a and any new position x. In essence, the agent moves towards the position on the disk where the total potential is minimized. Please note that the objective function $P_{total}(x, t)$ is non-linear, not differentiable, and discontinuous.

$$\min_{x \in \Omega} \quad P_{total}(x, t) \text{s.t.} \quad d_a(x) - r \leq 0 \tag{17}$$

2.4.2 Update scheme OSM model

The complete algorithm is depicted in Algorithm 5. The OSM update scheme is pretty straightforward. Per agent, the potential field is computed first. Accordingly, each agent is moved using the minimization problem. Each agent is only visited by the algorithm once per timestep. Consequently, also for this model type, a sequential update scheme is often applied to speed up the simulations.

Algorithm 5. OSM update scheme.

procedure OSMUPDATE(A, X)
  `Initialize` $A_{new} = A$ ▷ Set of agents to be moved
  while $A_{new} \neq \varnothing$ **do**
    $a \sim$ `Uniform`(A_{new}) ▷ Random select agent
    Determine current location x_p
    Determine obstacle distances d_o^a
    Determine agents distances d_p^a
    Compute potential fields $P_{obstacle}, P_{ped}, P_{total}$
    Solve the minimization problem x
    $X(a) \leftarrow x$ ▷ Set new location agent
    $A_{new} \leftarrow A_{new} \setminus \{a\}$ ▷ Remove a from A_{new}
  end while
  return X
end procedure

2.4.3 Recent OSM adaptations

A limited number of researchers have adopted the Optimal Steps Model. After its original presentation, it was validated by several studies, for instance, (Von Sivers and Köster, 2015; Seitz et al., 2015), and (Kneidl, 2016). (von Sivers et al., 2016) combined the OSM with the Social Identity Model

Application, a model that simulates the strategic goal-orientation behavior of agents during evacuations, to improve the simulation of evacuation scenarios in which social behavior might impact evacuation performance. (Köster et al., 2016) and (Zeng et al., 2018) extended the OSM to improve the behavior of agents on stairs. (Mayr and Köster, 2021) extended and re-calibrated the OSM for social distancing situations, as were enforced during the COVID-19 pandemic.

2.4.4 Benefits and challenges of the OSM model

The OSM has been calibrated and validated for several specific scenarios and implemented in the open-source simulation package Vadere (www.vader-e.org). As the OSM uses potential functions, one can simplify or extend the stepping behavior of agents as long as one can translate the interaction behavior between an agent and another entity (obstacle, individual, signal) into a potential. Consequently, similarly to the SF, it is possible to incorporate environment features (agents, objects, and guidance information) in the OSM.

At the same time, the interplay between potentials is fairly complex, which can result in unexpected artifacts. As agents orient themselves in relation to the floor field, which is the result of the movements of all surrounding signals, the computational effort of this model can be high. The procedures to compute the potential of the other pedestrians, which changes at every time step, are especially costly.

2.5 Discrete choice models

A fifth category of models was introduced by (Antonini et al., 2006), namely the discrete choice (DC) models. This model type assumes that agents are rational beings that optimize the utility of their choice, given a set of choice options. That is, pedestrians can compute and weigh the differences between potential future actions. In contrast to the previous microscopic simulation models, a discrete model's structure is calibrated completely using real data. Using an explicit error structure, it can also model the noise in pedestrians' choice behavior.

2.5.1 Mathematical formulation of the DC model

In this section, we present the first-ever discrete choice model specifically designed to capture the operational walking dynamics of pedestrians, namely the model of (Antonini et al., 2006). Since then, other applications for both pedestrians and cyclists have been introduced. Yet, this first application nicely captures the essence of the capabilities of the discrete choice model.

Within the DC model, time is discretized, and space is described continuously. Within this space, an agent makes choices using a discretized choice set. This choice set consists of a set of options that fan out the agent's current speed and orientation (see Fig. 3). This choice set is bounded by the vision angle of the agent, as well as their maximum acceleration and deceleration. Accordingly, alternatives within the choice space are generated by dividing the choice space into parts. In (Antonini et al., 2006), three acceleration regimes a_i (i.e., deceleration, neutral, acceleration) and eleven orientation regimes θ with varying apertures (i.e., 25 deg towards the outsides, 10 deg in the middle) are adopted. It is up to the modeler to balance the computational load with the required granularity of the choice set.

Accordingly, the utility of all alternatives is determined through a utility function (see Eq. (18)). In Antonini et al. (2006)'s model, the utility is expressed as a combination of the density at a specific location (occupation$_{p,v,\theta}$), the required change in the direction towards the pedestrians' goal (destination$_{p,\theta}$), their willingness to follow the existing flow pattern in a venue (angle$_{p,v,\theta}$), the angle with respect to the current speed (angle$_{p,v,\theta}$), and the cost of accelerating or decelerating.

$$V_{a,v,\theta} = \beta_{occ} * \text{occupation}_{a,v,\theta} + \beta_{dir} * \text{direction}_{a,\theta} + \beta_{dest} * \text{destination}_{a,\theta} + \beta_{angle} * \text{angle}_{a,v,\theta} + \beta_{acc} I_{v,acc} \left(\frac{v_{a,v}}{v_{a,max}}\right)^{\gamma_{acc}} + \beta_{acc} I_{v,dec} \left(\frac{v_{a,v}}{v_{a,max}}\right)^{\gamma_{dec}} \tag{18}$$

$$\text{occupation}_{a,v,\theta} = \sum_{n=1}^{N} I_{n,\theta} * e^{-\gamma_1 \| x_a - c_{v,\theta} \|} \tag{19}$$

where n represents the total number of pedestrian in the area, $I_{n,\theta}$ is a boolean identifying whether pedestrian n belongs to the directional cone of the alternative, $\|x_a - x_{v,\theta}\|$ the distance between pedestrian a and the center

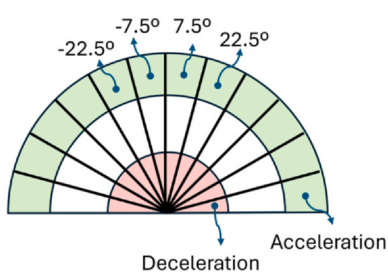

Fig. 3 Discretized choice set of movement options.

of the alternative $c_{a,v,\theta}$. γ_1 weights at what distance other pedestrians are considered when determining an alternative's occupation.

$$\text{angle}_{a,v,\theta} = \sum_{n=1}^{N} I_{n,\theta} * \alpha_{n,\theta} * e^{-\gamma_2 \| x_a - c_{v,\theta} \|} \tag{20}$$

Here, $\alpha_{n,\theta}$ accounts for the angle between the direction of agent a and each other agent n. The role of γ_2 is similar to that of γ_1, namely to calibrate the distance at which the directionality of surrounding agents is taken into account in this metric.

Afterward, a straightforward multinomial logit (MNL) modeling approach can be applied to derive the probability that a pedestrian will choose a specific alternative (see Eqs. (21) to (24)). Here, Eq. (21) computes the probability that alternative i is chosen given the set of alternatives C_a. Eqs. (22) and (23) accordingly update the speed v_a and direction $\vec{e}$ of the agent using the acceleration a_i and direction θ that are associated with alternative i. The last equation (Eq. (24)) moves agent a from its original location $x_a(t)$ to its new location $x_a(t + \delta t)$.

$$P(i|C_a) = \frac{e^{\mu V_{a,i}}}{\sum_{j \in C_a} e^{\mu V_{a,j}}} \tag{21}$$

$$v_a(t + \delta t) = v_a(t) + \delta t * a_i \tag{22}$$

$$\vec{e}_a(t + \delta t) = \vec{e}_a(t) + \vec{\theta} \tag{23}$$

$$x_a(t + \delta t) = x_a(t) + \delta t * v_a(t) * \vec{e}_a(t) \tag{24}$$

In discrete choice modelling, often, the errors are assumed to be independent and identically distributed (i.i.d.). Yet, the work of Antonini et al. (2006) assumes that the alternatives are not entirely independent of each other but are partly correlated. Therefore, in contrast to the regular MultiNomial Logit (MNL) formulation, they adopt a slightly more complex modeling structure, namely a cross-nested logit (CNL) modeling approach Bierlaire (2006) (see Eq. (25)). In this equation, $\alpha_{j,m}$ represents the membership to a nest, and μ is a scale parameter. In this particular case, five nests are identified: accelerating, constant speed, deceleration, central, and not central.

$$P(i|C) = \frac{\sum_m (\alpha_{i,m} \gamma_i)^{\mu_m} \left(\sum_j (\alpha_{i,m} \gamma_i)^{\mu_m}\right)^{\frac{\mu}{\mu_m} - 1}}{\sum_m \left(\sum_{j \in C} (\alpha_{i,m} \gamma_i)^{\mu_m}\right)^{\frac{\mu}{\mu_m}}} \tag{25}$$

2.5.2 Update scheme DC model

The whole updating procedure of the agent locations is displayed in Algorithm 6. Per agent, the set of alternatives is initialized. Accordingly, the utility of all alternatives is computed. Afterward, the choice probability of each alternative is determined. The agent is moved to a new location using the acceleration and direction associated with the alternative with the highest choice probability. Here, one can also draw one alternative using the computed choice probabilities to introduce stochasticity.

Algorithm 6. DC update scheme.

---

procedure DCUPDATE(A, X)
    `Initialize` $A_{new} = A$ ▷ Set of agents to be moved
    while $A_{new} \neq \varnothing$ **do**
        $a \sim$ `Uniform(`A_{new}`)` ▷ Random select agent
        Determine current location agent x_a
        Initialize choice set C_a
        Compute occupation $occupation_{a,v,\theta}$
        Compute angle $angle_{a,v,\theta}$
        Compute utility of alternatives $V_{a,v,\theta}$
        Compute probabilities alternatives $P(i|C_a)$
        Determine chosen alternative i
        Move agent to new location $x_a(t + \delta t)$
        $A_{new} \leftarrow A_{new} \setminus \{a\}$ ▷ Remove a from A_{new}
    end while
    return X
end procedure

Each agent is only visited by the algorithm once per timestep. Consequently, also for this model type, a sequential update scheme is often applied to speed up the simulations.

2.5.3 Recent DC model adaptations

The formulation of (Antonini et al., 2006) is just one particular formulation of the utility function. Depending on the scenario and/or context, additional elements can be added to this function to better capture pedestrians' movement dynamics and choice behavior.

Even though the model is very flexible, only a few enhancements of the naive model have been presented. (Robin et al., 2009) extended the utility function with variables to capture leader-following behavior and collision avoidance explicitly. (Fukuda et al., 2014) extends the econometric model by incorporating the pedestrians' destination and route as a latent variable, thus allowing dynamic changes to pedestrians' operational movement dynamics. (Gavriilidou et al., 2019) calibrated a discrete choice model to capture the movement dynamics of cyclists in a queue upstream of a traffic light. In addition (Gavriilidou et al., 2019) also observed that it is very challenging to transform the DC model into a comprehensive simulation environment.

2.5.4 Benefits and challenges of DC models

Discrete choice models are entirely data-driven, thus allowing researchers to use this model type to unravel pedestrian behavior more systematically. In essence, no insights into the underlying physical processes are required to incorporate a factor into the choice modeling structure. Consequently, it is also a flexible modeling structure that can quickly adapt to specific contexts and populations when data regarding their movement dynamics is readily available. Even though the DC models do not model forces directly, their continuous space interpretation allows users to determine friction forces in a crowd. Yet, we have not seen a study attempting to model high-density crowds with a discrete choice model.

Interestingly, the DC model's data-driven nature is also one of its main shortcomings. It has proven challenging to calibrate a generic DC model that functions well under varying conditions, most likely because not all relevant scenarios have sufficient datasets readily available.

2.6 Data-driven artificial neural network models

Data-driven Artificial Neural Network (ANN) models are a recent addition to the microscopic simulation model family tree. Through machine learning techniques, highly parameterized neural networks of various shapes, sizes, and structures are trained directly on big data sets featuring pedestrian trajectory data. In essence, the relationship between input (i.e., the distance of a pedestrian to surrounding obstacles, other pedestrians, and the direction of the goal) and output (i.e., the next step, speed, or acceleration of an agent) is captured using a parameterized set of interconnected simple linear equations.

There are a myriad of ANN models that simulate pedestrian choice behavior and movement dynamics. Most of these are used for pedestrian demand prediction (e.g., Cohen and Dalyot, 2020) and/or strategic route planning (e.g., Duives et al., 2019). Those endeavors are out of the scope of this chapter. In the remainder, we will focus on the operational movement prediction algorithms. In those cases, the ANNs predict individual agents' location, speed, and/or acceleration using machine learning approaches, particularly data-driven neural networks.

2.6.1 Mathematical formulation ANN models

In essence, neural networks are modeled to resemble a human brain. A net comprises a large number of simple processing nodes that are densely interconnected. The nodes are often arranged in layers and feedforward such that data flows through them in only one direction. Each node connects to a set of nodes in the layer beneath it, from which it receives data, and to a set of nodes in the layer above it, to which it transmits data. Depending on the link weights, the importance of the transmitted data is dampened or strengthened while moving from one layer to the next. In addition, data is only passed to the following layer when the signal is strong enough. No data is passed across the link to the next layer if the signal strength falls below a certain threshold (Fig. 4).

The mathematical formulation of the process within each neuron of a feedforward network is relatively simple (see Eq. (26)). In essence, the output of neuron n in layer l is the result of the data $x_i(l-1)$ received to all neurons n of the previous layer $l-1$ multiplied with the weights $w_i(l-1)$ of all the incoming links. Biases $b_n(l)$ are often applied to improve the learning and accuracy of the trained neural network.

$$y_n(l) = f\left(b_n(l) + \sum_{i=1}^{n} x_i(l-1)\, w_i(l-1)\right) \tag{26}$$

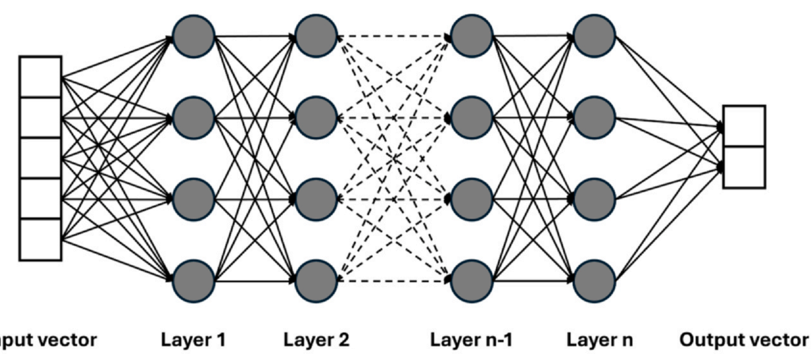

Fig. 4 Visual impression of a simple neural network.

The function f represents the type of decision-making within the node, the so-called activation function. Three common activation functions are:

- **Rectified Linear Units (ReLU)** $f(z) = (max)(0, z)$ The output remains larger than zero and is scaled linearly.
- **Tanh** $f(z) = (tanh)(z)$ The output is the hyperbolic tangent of z, which rescales the output values to values between -1 and 1 with relatively steep gradients around zero.
- **sigmoid** $f(z) = 1/(1 + e^{-z})$ The output is the hyperbolic tangent of z, which rescales the output values to values between 0 and 1. The gradients around 0 are flatter than the tanh function.
- Gaussian Error Linear Unit (GeLU) $f(z) = zP(X \leq x) = x\Phi(x)$ The output scales linearly with the input for values above zero and remains approximately 0 for values under 0. Yet, the gradient transition around zero is less sharp than in a ReLU.

An essential property of a feedforward neural network is its lack of memory. Feedforward ANNs only react directly to the input provided at a given timestep. Consequently, it has difficulties capturing time-based trends in data, where data from previous timesteps is correlated to the data of the current timestep. To enable an ANN to "remember" data of earlier timesteps, more complex node types have been developed, among which are the Gated Recurrent Units (GRU), Long Short-Term Memory (LSTM), and Self-Attention. The first two apply a gating mechanism that retains or forgets certain features of the data from previous time steps. However, the order of the information retention never changes. In a self-attention layer, the output weights depend on the current element in the sequence the model is focusing on. Consequently, the self-attention layer allows a network model to focus on the most relevant parts of the input sequence based on the attention weights, which can vary per input sequence element.

Lastly, the architecture of the neural network is increasingly part of the design of neural networks. Early modeling attempts featured predominantly fully connected networks featuring one node type. More recent ANN architectures combine the properties of multiple node types and leverage the interesting properties (e.g., faster learning, less memory usage, improved generalization) of sparse connected networks. To date, the convolutional neural network (CNN) and the Large Language Model (LLM) are interesting network architectures that can be leveraged for pedestrian movement dynamics prediction. The CNN is a neural network type where at least one hidden layer is not strictly fully connected to the previous and following layers of the network. Thus, convolutional networks can extract features from the input data and are often applied to computer vision tasks. The LLM is an ANN model initially designed for natural language processing tasks. These neural networks with millions of neurons are composed of varying types of neural network layers, such as feedforward, embedding, recurrent, and attention layers. This last model architecture especially has great potential with regard to the modeling of pedestrian movements.

The developments regarding neural network architecture design move incredibly quickly in this day and age. If you are interested in the most recent ANN models, you are advised to also search the internet for more recent ANN model architecture developments.

2.6.2 Update scheme ANN model

The update procedure of an ANN depends heavily on the ANN model type, the exact implementation of the ANN model, and the exact input/output of the ANN model. Therefore, providing a generic description of the ANN update scheme is challenging. The reader is advised to follow the update schemes mentioned in the papers featuring ANN models that simulate pedestrian movement dynamics.

2.6.3 Recent ANN model adaptations

In the past few years, the number of developed data-driven, fully connected feedforward neural networks, also called MultiLayer Perceptrons (MLP), has risen rapidly. Among others, (Kielar and Borrmann, 2020; Tordeux et al., 2019), and (Ma et al., 2019) developed a relatively straightforward neural network where the inputs feature the pedestrian's perceived environmental information and the outputs of the agent's walking behavioral response. Some also tested other input information. (Song et al., 2018), for instance, normalizes the input layer relative to the pedestrian and incorporates more path-planning information in the input layer. (Sun et al., 2023), on the other hand, uses physical properties (e.g., time to collision and the agent's acceleration) and social dynamics (e.g., inter-group relationships) to improve the model's ability to capture pedestrian walking dynamics.

Some others adopt multiple neural networks simultaneously to inform a physics model. For example, (Hossain et al., 2022) combine a social force model extended by a group force with a Multilayer Perceptron. In essence, four physical forces are estimated separately and combined using the general mathematical equations of the Social Force model. The ANN inputs resemble the inputs of the naive general social force model, yet this ANN does not explicitly define the mathematical relation between the inputs and outputs (i.e., forces).

Besides that, researchers are increasingly developing neural networks with divergent architectures. Among other studies, (Alahi et al., 2016) used an LSTM to predict human-human interactions. More recently, (Wang et al., 2024) adopted a Convolutional Neural Network (CNN). More extensive modeling attempts are expected to be presented in the following years when the interesting properties of MLP, CNN, RNN, and LSTM are combined. Recent LLM modeling attempts provide a nice example of the potential of such combinations.

2.6.4 Benefits and challenges of ANN models

The main benefit of ANN models is that they do not explicitly specify the relationships between input and output variables. Consequently, as long as one has a sufficiently big dataset, one can capture the trends present within it. Insights into the trends and features of the data help modelers make informed choices regarding the most applicable ANN architecture. In addition, new training schemes, such as online learning, federated learning, and distributed learning, allow modelers to create adaptable and versatile models.

There are also some challenges regarding the training and deployment of ANN models. First and foremost, the quality of the training data severely impacts the models' prediction capabilities. In general, large quantities of diverse (i.e., capturing all movement base cases), balanced (e.g., varying goal orientations, group formations, density levels), and unbiased (e.g., diverse w.r.t. age, gender, race, culture) trajectories are required to develop a generic microscopic pedestrian model. Secondly, comprehensive ANN-based microscopic pedestrian models generally feature a huge parameter set. Thus, the computational efforts of training and deploying an ANN can be extensive. Similarly to any other model type, this currently limits their use for large-scale crowd movement prediction. Lastly, any ANN is trained to fit the average behavior in the training data. This currently limits the applicability of ANNs for scenarios outside the ordinary, such as large-scale evacuations, sudden service disruptions, and ambiguous scenarios where social interactions matter.

3. When to use which model type?

To apply models correctly, it is essential to use the appropriate model type for each model application. It is important to realize that most microscopic models are built with a specific goal and are not designed to capture the whole range of pedestrian crowd movement dynamics. Consequently, we must ensure that a model can capture the pedestrian movement dynamics that can present themselves in a given flow scenario and context. In Section 3.4, a comparison between the model types is presented.

Before going into the comparison, some background knowledge regarding operational movement behaviors we try to capture in microscopic pedestrian simulation models is required. The literature on operational movement dynamics identifies two types of operational movement dynamics that we must consider when determining which model to apply: the movement-based cases and emerging self-organization phenomena. In the following three sections, we will first present the definition of the movement base cases (Section 3.1), the self-organization phenomena (Section 3.2) and the mathematical properties required to model a movement base case and/or a self-organization phenomenon (Section 3.3).

3.1 Movement base cases

The movement base cases, also called flow scenarios in the literature, were initially proposed by (Duives et al., 2013), and provide a topology of the distinct actions a crowd can perform within an infrastructure (see Fig. 5). Jointly, eight distinct movement base cases cover the whole range of pedestrian operational movement dynamics.

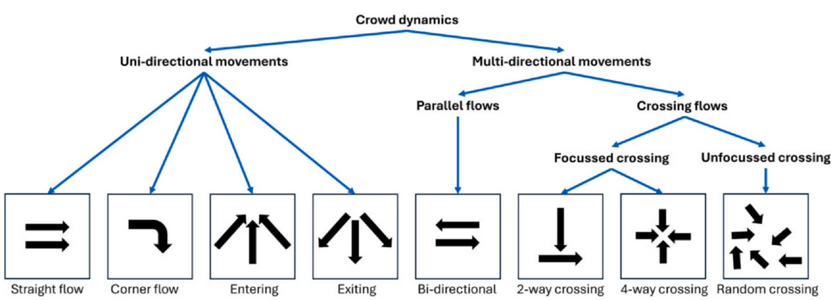

Fig. 5 Topology movement base cases.

The topology first distinguishes between unidirectional and multi-directional flows. The first category of movement-based cases describes unidirectional interactions, where all agents move in the same direction. The second category features interaction between individuals under an angle, where agents must more proactively avoid collisions. Unidirectional flows can either be straight flows (i.e., no changes in available space or changes in direction), flows rounding corners (i.e., a change in direction, but no need for compaction or expansion), flows entering a bottleneck (i.e., a decrease of available walkway space resulting in compaction of the crowd) or flows exiting a bottleneck (i.e., an increase of available walking space, resulting in expansion of the crowd). Multi-directional flows interact under an angle, either in a parallel counter-flow (i.e., bi-directional flow featuring friction on the flow direction interfaces) or under an angle. The exact movement pattern for crossing flows depends on the available space and the number of flow directions in a particular part of the infrastructure. Here, three dominant scenarios arise. The first and second are crossings with a focal point of the interactions, where two or more flow directions intersect in a very confined space (e.g., the intersection of two corridors in a metro station). The third scenario is a random crossing situation, where pedestrians experience a lot of consecutive interactions with different individuals (i.e., random crossing at a large square).

3.2 Self-organization patterns

In addition to the movement base cases, distinct types of emergent movement patterns also arise in a crowd. These so-called self-organization patterns are defined in literature as the spontaneous establishment of qualitatively new behavior through the non-linear interaction of many objects or subjects (Helbing and Johansson, 2010) without the intervention of external influences (Camazine et al., 2020). Similarly to the movement base cases, the mathematical properties required to simulate a particular self-organization pattern differ. Consequently, we can use the self-organization patterns to compare the microscopic pedestrian simulation model types.

In literature, six self-organization phenomena have been described for pedestrian crowds: lane formation, Stop&Go waves, herding, the zipper effect, freezing by heating, and turbulence. During lane formation, lanes of varying width form dynamically in a corridor (Hoogendoorn and Daamen, 2005). Stop & Go waves are temporarily interrupted longitudinal flows that appear at higher densities in unidirectional crowds (Helbing et al., 2007). Herding describes a situation where pedestrians follow other pedestrians' movement direction without explicit instructions because they expect others to know more than they do (Helbing, 2012). The zipper effect describes how pedestrians diagonally allow other pedestrians within their personal space in front of them for a short period of time to pass a narrow bottleneck (Hoogendoorn and Daamen, 2005). Freezing-by-heating emerges when coordination problems occur just upstream of the bottleneck, forcing the crowd to a standstill due to many individuals competing for the same space (Helbing and Johansson, 2010). Turbulence describes an emergent pattern where a granular flow displays fluid-like properties, and pedestrians no longer control their movements (Helbing et al., 2007).

3.3 Properties required to simulate movement base cases and self-organization patterns

The previous section shows that each microscopic pedestrian simulation model type has varying capabilities. Yet, most types of microscopic simulation models cannot model all movement base cases and self-organization patterns. Consequently, it is essential to adopt a model type that can simulate the movement dynamics and self-organization patterns that one can expect in the context that one is modeling.

Movement-based cases and self-organization phenomena are still infrequently or inconsistently used to validate a model type. Therefore, we must first establish which properties a model type requires to simulate movement base cases and/or self-organization patterns. We do so by establishing which mathematical properties are needed to simulate particular movement dynamics and/or patterns.

The pedestrian simulation literature suggests that a limited set of model characteristics is responsible for the movement base cases and self-organization patterns. These characteristics are mentioned below.

Collision avoidance - Literature shows that people start adjusting their paths several meters before reaching a conflict to reduce the interaction and to avoid collisions with other pedestrians (Goffman, 2017). Therefore, models that incorporate rules to avoid collision with other (moving) agents are assumed to model collision avoidance. Collision avoidance is essential to realistically model multi-directional movement base cases (i.e., bi-directional and crossing movements). Here, we must distinguish between model types that can only capture very local collision avoidance actions (e.g., CA model) and those that can resolve potential collisions at a distance (a.o., SF, PCA, and OSM models).

Reaction time – The presence of reaction time in a model slows the reaction of pedestrians to new traffic situations. The pedestrians adapt their speed slowly, thereby overshooting the optimal solution and magnifying the occurring traffic flow state. We assume Stop&Go waves can emerge when, in the mathematical formulation, a reaction/adaptation time is specified for any velocity and acceleration increase and/or decrease.

Dynamic changes of personal space definition – Inside shallow bottlenecks, individuals allow for a temporary decrease in their personal space. If a model can simulate location-specific changes in the infrastructure, we assume the model can capture the zipper. Often, model types allow for a decrease in personal space or the repulsive forces at a specific location within the infrastructure.

Force transfer at a distance – A funnel-like shape appears when pedestrians exit a bottleneck. This funnel-like shape can only be recreated when a model explicitly pushes agents away from each other while already physically separated completely. Thus, models incorporating forces/pressure emitted by other pedestrians and/or obstacles at a distance can model the exiting movement base case.

Force transfer between agents (and obstacles) – When pedestrians push each other or touch while moving past each other in crowded scenarios, forces are transferred from one person to the other. Models that explicitly simulate these interpersonal local force-based interactions can capture friction and, thus, turbulent fluid-like movements. In addition, friction and bodily contact are also essential in modeling the friction in a crowd upstream of a bottleneck, which causes the freezing-by-heating effect.

Compresion of agents – In a very dense crowd, pressure can be built up due to forces being transferred from person to person, resulting in pressure points in a crowd (Challenger et al., 2009). Some models, explicitly or implicitly, allow for the compression of the pedestrian body, thereby allowing high densities to emerge in crowded scenarios. Models that allow for the compression of the pedestrian body can model pressure built up in turbulent scenarios.

Strategic route changes due to local dynamics – During a herding situation, the goal orientation of pedestrians changes due to the behaviors of surrounding pedestrians. Thus, to model herding, a model should incorporate explicit rules on dynamic adaptations of agents' strategic route choice behavior due to local changes in movement dynamics. In essence, any microscopic model can handle dynamic changes in the strategic choice behavior of agents. Consequently, all six model types can potentially capture this property when calibrated correctly.

3.4 Model type comparison

Some general characteristics of the six model types are summarized in Table 1. Here, the 'obstacles' and 'agents' columns identify at what distance obstacles and other agents are considered in an agent's movement computation.

The capabilities of each model type can be assessed using the descriptions of the model types in Section 2, the properties required to simulate microscopic pedestrian movement dynamics, and the characteristics mentioned in Table 1. Please note that the model types are not only evaluated based on each model type's most naive model description. Instead, potential enhancements of the mathematical and computational architecture of the model category are also considered. Consequently, a model type is not necessarily limited to the specific application for which the model was initially developed.

The comparison in Table 2 shows that most models can capture most movement base cases. Unidirectional and bidirectional straight movements can be captured by all model types, which is unsurprising because all models are explicitly designed to capture these two behaviors. The only two movement base cases that almost all model types have difficulties with is the expansion behavior of a crowd after a bottleneck. Most model types do not explicitly model the forces required to simulate the expansion that occurs just downstream of a bottleneck behavior explicitly. As a result, most model types will overestimate the densities and underestimate the speeds downstream of a bottleneck. Besides that, most model types might produce modeling artifacts just upstream of a bottleneck due to the presence of wall forces and/or potentials that point in the opposite direction of the agents' movement velocity, thereby slowing down the agents to an unrealistic extent. This can potentially result in the clogging of bottlenecks in scenarios where one would not see clogging in reality.

At the same time, most model types cannot simulate a large part of the self-organization phenomena. Self-organization phenomena that emerge in high-density crowds, such as freezing-by-heating and turbulence, are especially challenging for most microscopic pedestrian simulation model types. Partly, this is because the datasets featuring this type of movement dynamics are not readily available to calibrate for this type of movement dynamics. Partly, this is due to the lack of mathematical constructions that can explicitly account for high-density crowd movements. Consequently, when modeling high-density scenarios, one must carefully check whether

Table 1 Summary of main characteristics of microscopic model types, where CA = Cellular Automata, SF = Social Force model, PCA = Predictive Collision Avoidance, OSM = Optimal Step Model, DC = Discrete choice model and ANN = Artificial Neural Network model.

Model type	Time	Space	Choice	Obstacles	Agents
CA	Discrete	Discrete	Discrete	Neighbours	Neighbours
SF	Discrete	Continuous	Continuous	Region	Region
PVA	Discrete	Continuous	Discrete	Vision field	Vision field
OSM	Discrete	Continuous	Continuous	Region	Region
DC	Discrete	Continuous	Discrete	Vision field	Vision field
ANN	Discrete	Continuous	Discrete	Region	Region

Table 2 Comparison of microscopic model types, where CA = Cellular Automata, SF = Social Force model, PCA = Predictive Collision Avoidance, OSM = Optimal Step Model, DC = Discrete choice model and ANN = Artificial Neural Network model, + = this model type can predict this behavior, − = this model type cannot predict this behavior, +/− = it depends on the exact formulation whether the model can predict this behavior, ? = There is potential for future enhancements to this model type to simulate this behavior, but no implementation has yet been presented to the authors' knowledge.

	CA	SF	PCA	OSM	DC	ANN
MBCs						
Uni-directional	+	+	+	+	+	+
Corners	+/−	+	+	+	+	?
Entering	+	+	+	+	+	?
Exiting	−	+	−	−	−	?
Bi-directional	+	+	+	+	+	+
Intersecting	+/−	+	+	+	+	+
Self-organisation phenomena						
Lane-formation	+	+	+	+	+	+
Stop & Go waves	−	+	−	+	−	?
Herding	+	+	+	+	+	+
Zipper-effect	−	+	+	+	+	?
Freezing-by-heating	−	+	−	+/−	−	?
Turbulence	−	+	−	+	−	?
Pressure built-up	−	+	−	+/−	−	?

the results of a microscopic pedestrian simulation model align with one's expectations and data featuring pedestrians' choice behavior and walking dynamics under these more challenging conditions.

What is essential to keep in mind when starting a new simulation study is that the best model for each job is dependent on the choice behavior, movement dynamics, and context one is trying to model. It is imperative that at the start of any new project, one makes a conscious decision on which model type and particular model realization one uses. It is a good exercise for any modeler to explicitly write down which behaviors one needs to capture realistically and accordingly check the functional description of the model and the manual of the simulation software one is using, whether the model can reproduce these behaviors in the context one is going to model.

4. Miscellaneous remarks

There are three other essential things to remember when modeling pedestrians' movement dynamics at a microscopic level.

Firstly, the performance of the model types above also severely depends on the behavioral processes that form an input to the model. Understanding the demand patterns, activity scheduling patterns, and population profiles (i.e., free-flow speed, acceleration abilities, weight, circumference, and presence of baggage) is essential to creating a realistic pedestrian simulation. Therefore, it is imperative to perform a site visit and thoroughly question your client to check any generic assumptions you have made in this regard.

Secondly, most simulation models do not assume social interactions between the agents that make up a crowd. Yet, the work of, among others Irnich et al. (2022), identify that large differences in overall evacuation performance can arise due to the social interactions between leaders and followers in a crowd. Similarly, Von Krüchten and Schadschneider (2017) shows that grouping in crowds impacts evacuation performance. This does not necessarily mean one cannot model without accounting for the social interactions. However, one has to be aware that ignoring social interactions can lead to an under- or overestimation of the overall performance of crowds.

Thirdly, the focus of this chapter has been on the capabilities and possible extensions of the current microscopic pedestrian models featuring operational movement dynamics. The precision of the models has not been a point of attention in this chapter, but it is of the utmost importance when implementing any pedestrian simulation model. For all model types, the actual functioning of a model type severely depends on the correct calibration and validation of the simulation model for each specific case. The reader is referred to chapter 9 for a thorough discussion of the options to enhance the quality of the predictions through calibration and validation and chapter 10 for a discussion of relevant model application guidelines.

5. Summary

This chapter introduced a variety of microscopic model types currently used to model pedestrian movement dynamics at a microscopic level, including the Cellular Automata model (CA), the Social Force model (SF), the predictive collision avoidance model (PCA), the Optimal Step Model (OSM), the Discrete Choice model (DC), and the data-driven artificial neural network model (ANN).

The overview shows that the resulting movement dynamics of each agent result from the interactions between agents, their physical and physiological environment, and available information. Each model type simulates the movement dynamics of agents in a particular way, resulting in slightly different operational and global movement dynamics. This chapter has presented the most naive version of each model type, as this best describes the agents' general choice behavior and resulting movement dynamics. In addition, some interesting extensions, benefits, and challenges of each model type were discussed.

A comparison of model types shows that most models can capture most movement base cases, also called flow scenarios. At the same time, most model types cannot simulate a substantial number of self-organization phenomena. Partly, this is because the datasets featuring this type of movement dynamics are not readily available to calibrate for this type of movement dynamics. Partly, this is due to the lack of mathematical constructions that can explicitly account for (high-density) self-organization phenomena, such as freezing-by-heating and turbulence.

At the same time, microscopic simulation models remain a simplification of reality. Consequently, the performance of the model types above also severely depends on the behavioral processes that form an input to the model. In addition, one has to be aware that the current assumptions regarding the limited social interactions between agents can lead to an under- or overestimation of the overall crowd dynamics. Lastly, the actual functioning of a model type severely depends on the correct calibration and validation of the simulation model for each specific case.

Acknowledgements

This work is part of the VENI project CrowdIT Space project that was funded by Dutch Research Council (NWO) no. 18083.

Ethics

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Dorine C. Duives: Conceptualization, Methodology, Formal analysis, Data curation, Writing - Original Draft, Writing - Review & Editing, Funding Acquisition, Supervision.

References

Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S., 2016. Social lstm: human trajectory prediction in crowded spaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 961–971.

Antonini, G., Bierlaire, M., Weber, M., 2006. Discrete choice models of pedestrian walking behavior. Transp. Res. B: Methodol. 40 (8), 667–687.

Bandini, S., Rubagotti, F., Vizzari, G., Shimura, K., 2011. A cellular automata model for pedestrian and group dynamics. In: WOA, pp. 34–40.

Bazior, G., Pałka, D., Was, J., 2020. Using cellular automata to model high density pedestrian dynamics. International Conference on Computational Science. Springer, pp. 486–498.

Bierlaire, M., 2006. A theoretical analysis of the cross-nested logit model. Ann. Oper. Res. 144, 287–300.

Blue, V.J., Adler, J.L., 1998. Emergent fundamental pedestrian flows from cellular automata microsimulation. Transp. Res. Rec. 1644 (1), 29–36.

Blue, V.J., Adler, J.L., 1999. Cellular automata microsimulation of bidirectional pedestrian flows. Transp. Res. Rec. 1678 (1), 135–141.

Blue, V.J., Adler, J.L., 2000. Modeling four-directional pedestrian flows. Transp. Res. Rec. 1710 (1), 20–27.

Camazine, S., Deneubourg, J.-L., Theraula, G., Sneyd, J., Franks, N.R., 2020. Selforganization in biological systems. Princeton Studies in Complexity. Princeton University Press. ISBN: 9780691116242.

Challenger, R., Clegg, C.W., Robinson, M.A., Leigh, M., 2009. Understanding crowd behaviours: supporting evidence. Understanding Crowd Behaviours'(Crown, 2009) 1–326.

Chen, C., Lu, T., 2024. An extended model for crowded evacuation considering stampede on inclined staircases. Simul. Model. Pract. Theory, 102978.

Chen, X., Treiber, M., Kanagaraj, V., Li, H., 2018. Social force models for pedestrian traffic–state of the art. Transp. Rev. 38 (5), 625–653.

Cohen, A., Dalyot, S., 2020. Machine-learning prediction models for pedestrian traffic flow levels: towards optimizing walking routes for blind pedestrians. Trans. GIS 24 (5), 1264–1279.

Cristiani, E., Frasca, P., Piccoli, B., 2011. Effects of anisotropic interactions on the structure of animal groups. J. Math. Biol. 62 (4), 569–588.

Daamen, W., 2002. Simped: a pedestrian simulation tool for large pedestrian areas. In: Congres EuroSIW, pp. 1–11. Simulation Interoperability Standards Organization.

Dias, C., Abdullah, M., Sarvi, M., Lovreglio, R., Alhajyaseen, W., 2019. Modeling and simulation of pedestrian movement planning around corners. Sustainability 11 (19), 5501.

Dias, C., Lovreglio, R., 2018. Calibrating cellular automaton models for pedestrians walking through corners. Phys. Lett. A 382 (19), 1255–1261.

Duives, D.C., Daamen, W., Hoogendoorn, S.P., 2013. State-of-the-art crowd motion simulation models. Transp. Res. C: Emerg. Technol. 37, 193–209.

Duives, D.C., Wang, G., Kim, J., 2019. Forecasting pedestrian movements using recurrent neural networks: an application of crowd monitoring data. Sensors 19 (2), 382.

Fukuda, D., Seo, T., Yamada, K., Yaginuma, H., Matsuyama, N., 2014. An econometric-based model of pedestrian walking behavior implicitly considering strategic or tactical decisions. Pedestrian and Evacuation Dynamics 2012. Springer, pp. 615–624.
García, A., Hernández-Delfin, D., Lee, D.-J., Ellero, M., 2023. Limited visual range in the social force model: effects on macroscopic and microscopic dynamics. Phys. A: Stat. Mech. Appl. 612, 128461.
Gavriilidou, A., Daamen, W., Yuan, Y., Hoogendoorn, S., 2019. Modelling cyclist queue formation using a two-layer framework for operational cycling behaviour. Transp. Res. C: Emerg. Technol. 105, 468–484.
Goffman, E., 2017. Relations in Public: Microstudies of the Public Order. Routledge.
Guy, S.J., Chhugani, J., Kim, C., Satish, N., Lin, M., Manocha, D., Dubey, P., 2009. Clearpath: highly parallel collision avoidance for multi-agent simulation. In: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 177–187.
Helbing, D., 2012. Self-organization in pedestrian crowds. Social Self-Organization: Agent-Based Simulations and Experiments to Study Emergent Social Behavior. Springer, pp. 71–99.
Helbing, D., Johansson, A., 2010. Cooperation, norms, and revolutions: a unified game-theoretical approach. PloS ONE 5 (10), e12530.
Helbing, D., Johansson, A., Al-Abideen, H.Z., 2007. Dynamics of crowd disasters: an empirical study. Phys. Rev. E—Stat. Nonlin. Soft Matter Phys. 75 (4), 046109.
Helbing, D., Molnar, P., 1995. Social force model for pedestrian dynamics. Phys. Rev. E 51 (5), 4282.
Heliövaara, S., Korhonen, T., Hostikka, S., Ehtamo, H., 2012. Counterflow model for agent-based simulation of crowd dynamics. Build. Environ. 48, 89–100.
Hoogendoorn, S., Daamen, W., 2005. Self-organization in pedestrian flow. Traffic and Granular Flowa'03. Springer, pp. 373–382.
Hoogendoorn, S.P., Bovy, P.H., 2004. Pedestrian route-choice and activity scheduling theory and models. Transp. Res. B: Methodol. 38 (2), 169–190.
Hoogendoorn, S.P., van Wageningen-Kessels, F.L., Daamen, W., Duives, D.C., 2014. Continuum modelling of pedestrian flows: from microscopic principles to self-organised macroscopic phenomena. Phys. A: Stat. Mech. Appl. 416, 684–694.
Hossain, S., Johora, F.T., Müller, J.P., Hartmann, S., Reinhardt, A., 2022. Sfmgnet: a physics-based neural network to predict pedestrian trajectories. arXiv Preprint arXiv:2202.02791.
Hu, J., Li, Z., Zhang, H., Wei, J., You, L., Chen, P., 2015. Experiment and simulation of the bidirectional pedestrian flow model with overtaking and herding behavior. Int. J. Mod. Phys. C 26 (11), 1550131.
Hu, J., You, L., Zhang, H., Wei, J., Guo, Y., 2018. Study on queueing behavior in pedestrian evacuation by extended cellular automata model. Phys. A: Stat. Mech. Appl. 489, 112–127.
Hughes, R.L., 2002. A continuum theory for the flow of pedestrians. Transp. Res. B: Methodol. 36 (6), 507–535.
Huo, F., Deng, S., Guo, C., Ma, Y., Zhang, W., 2024. Evacuation simulation considering pedestrian risk perception under toxic gas diffusion. Int. J. Disaster Risk Reduct., 104589.
Irnich, J., Wal, N.V.D., Duives, D., Auping, W., 2022. Impact of leader-follower behavior on evacuation performance: an exploratory modeling approach. Conference of the European Social Simulation Association. Springer, pp. 153–165.
Johansson, F., Peterson, A., Tapani, A., 2015. Waiting pedestrians in the social force model. Phys. A: Stat. Mech. Appl. 419, 95–107.

Karamouzas, I., Overmars, M., 2010. A velocity-based approach for simulating human collision avoidance. Intelligent Virtual Agents: 10th International Conference, IVA 2010, Philadelphia, PA, USA, September 20-22, 2010. Proceedings 10. Springer, pp. 180–186.

Kielar, P., Borrmann, A., 2020. An artificial neural network framework for pedestrian walking behavior modeling and simulation. Coll. Dyn. 5, 290–298.

Kim, S., Guy, S.J., Hillesland, K., Zafar, B., Gutub, A.A.-A., Manocha, D., 2015. Velocity-based modeling of physical interactions in dense crowds. Vis. Comput. 31, 541–555.

Kneidl, A., 2016. How do people queue? A study of different queuing models. Traffic and Granular Flow'15. Springer, pp. 201–208.

Köster, G., Lehmberg, D., Dietrich, F., 2016. Is slowing down enough to model movement on stairs? Traffic and Granular Flow'15. Springer, pp. 35–42.

Lee, J., Kim, T., Chung, J.-H., Kim, J., 2016. Modeling lane formation in pedestrian counter flow and its effect on capacity. KSCE J. Civil Eng. 20, 1099–1108.

Ma, Y., Lee, E.W., Hu, Z., Shi, M., Yuen, R.K., 2019. An intelligence-based approach for prediction of microscopic pedestrian walking behavior. IEEE Trans. Intell. Transp. Syst. 20 (10), 3964–3980.

Ma, Y., Zhang, G., Huo, F., 2024. An evacuation model considering pedestrian fall behavior in an inclined passenger ship. Phys. A: Stat. Mech. Appl. 643, 129794.

Mayr, C.M., Köster, G., 2021. Social distancing with the optimal steps model. Coll. Dyn. 6, 1–24.

Moussaïd, M., Perozo, N., Garnier, S., Helbing, D., Theraulaz, G., 2010. The walking behaviour of pedestrian social groups and its impact on crowd dynamics. PloS ONE 5 (4), e10047.

Paris, S., Pettré, J., Donikian, S., 2007. Pedestrian reactive navigation for crowd simulation: a predictive approach. Comput. Graph. Forum 26 (3), 665–674.

Parisi, D.R., Gilman, M., Moldovan, H., 2009. A modification of the social force model can reproduce experimental data of pedestrian flows in normal conditions. Phys. A: Stat. Mech. Appl. 388 (17), 3600–3608.

Pellegrini, S., Ess, A., Schindler, K., Van Gool, L., 2009. You'll never walk alone: modeling social behavior for multi-target tracking. In: 2009 IEEE 12th international conference on computer vision, IEEE, pp. 261–268.

Qu, Y., Gao, Z., Xiao, Y., Li, X., 2014. Modeling the pedestrian's movement and simulating evacuation dynamics on stairs. Saf. Sci. 70, 189–201.

Ren, Z., Charalambous, P., Bruneau, J., Peng, Q., Pettré, J., 2017. Group modeling: a unified velocity-based approach. Comput. Graph. Forum 36 (8), 45–56.

Robin, T., Antonini, G., Bierlaire, M., Cruz, J., 2009. Specification, estimation and validation of a pedestrian walking behavior model. Transp. Res. B: Methodol. 43 (1), 36–56.

Schadschneider, A., Kirchner, A., Nishinari, K., 2002. Ca approach to collective phenomena in pedestrian dynamics. International Conference on Cellular Automata. Springer, pp. 239–248.

Seitz, M.J., Dietrich, F., Köster, G., 2015. The effect of stepping on pedestrian trajectories. Phys. A: Stat. Mech. Appl. 421, 594–604.

Seitz, M.J., Köster, G., 2012. Natural discretization of pedestrian movement in continuous space. Phys. Rev. E 86 (4), 046108.

Seyfried, A., Steffen, B., Lippert, T., 2006. Basics of modelling the pedestrian flow. Phys. A: Stat. Mech. Appl. 368 (1), 232–238.

Song, X., Han, D., Sun, J., Zhang, Z., 2018. A data-driven neural network approach to simulate pedestrian movement. Phys. A: Stat. Mech. Appl. 509, 827–844.

Sun, S., Sun, C., Duives, D.C., Hoogendoorn, S.P., 2023. Neural network model for predicting variation in walking dynamics of pedestrians in social groups. Transportation 50 (3), 837–868.

Tordeux, A., Chraibi, M., Seyfried, A., Schadschneider, A., 2019. Prediction of pedestrian speed with artificial neural networks. Traffic and Granular Flow'17 12. Springer, pp. 327–335.

Van Den Berg, J., Patil, S., Sewall, J., Manocha, D., Lin, M., 2008. Interactive navigation of multiple agents in crowded environments. In: Proceedings of the 2008 symposium on Interactive 3D graphics and games, pp. 139–147.

Von Krüchten, C., Schadschneider, A., 2017. Empirical study on social groups in pedestrian evacuation dynamics. Phys. A: Stat. Mech. Appl. 475, 129–141.

VonSivers, I., Köster, G., 2015. Dynamic stride length adaptation according to utility and personal space. Transp. Res. B: Methodol. 74, 104–117.

von Sivers, I., Templeton, A., Künzner, F., Köster, G., Drury, J., Philippides, A., Neckel, T., Bungartz, H.-J., 2016. Modelling social identification and helping in evacuation simulation. Saf. Sci. 89, 288–300.

Wang, T., Zhang, Z., Nong, T., Tan, J., Lan, W., Zhang, W., Lee, E.W.M., Shi, M., 2024. Modelling the unidirectional and bidirectional flow of pedestrians based on convolutional neural networks. Phys. A: Stat. Mech. Appl. 651, 130021.

Xie, C.-Z., Tang, T.-Q., Hu, P.-C., Chen, L., 2022. Observation and cellular-automaton based modeling of pedestrian behavior on an escalator. Phys. A: Stat. Mech. Appl. 605, 128032.

Xu, Q., Chraibi, M., Seyfried, A., 2021. Anticipation in a velocity-based model for pedestrian dynamics. Transp. Res. C: Emerg. Technol. 133, 103464.

Xu, S., Duh, H.B.-L., 2009. A simulation of bonding effects and their impacts on pedestrian dynamics. IEEE Trans. Intell. Transp. Syst. 11 (1), 153–161.

Yu, W., Johansson, A., 2007. Modeling crowd turbulence by many-particle simulations. Phys. Rev. E—Stat. Nonlin. Soft Matter Phys. 76 (4), 046105.

Yuen, J., Lee, E., 2012. The effect of overtaking behavior on unidirectional pedestrian flow. Saf. Sci. 50 (8), 1704–1714.

Zeng, Y., Song, W., Huo, F., Vizzari, G., 2018. Modeling evacuation dynamics on stairs by an extended optimal steps model. Simul. Model. Pract. Theory 84, 177–189.

Zhang, J., Cui, H., Chraibi, M., Yu, H., Song, W., 2023a. Velocity-based model for pedestrian dynamics considering direction preferences. Phys. A: Stat. Mech. Appl. 613, 128444.

Zhang, Q., Qu, J., Han, Y., 2023b. Pedestrian small group behaviour and evacuation dynamics on metro station platform. J. Rail Transp. Plan. Manag. 26, 100387.

Zheng, Z., Zhu, G., Sun, Z., Wang, Z., Li, L., 2020. Improved social force model based on emotional contagion and evacuation assistant. IEEE Access 8, 195989–196001.