Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2014)
Vladlen Koltun and Eftychios Sifakis (Editors)

DenseSense: Interactive Crowd Simulation using Density-Dependent Filters

A. Best and S. Narang and S. Curtis and D. Manocha

Department of Computer Science
University of North Carolina at Chapel Hill
http://gamma.cs.unc.edu/DenseSense/

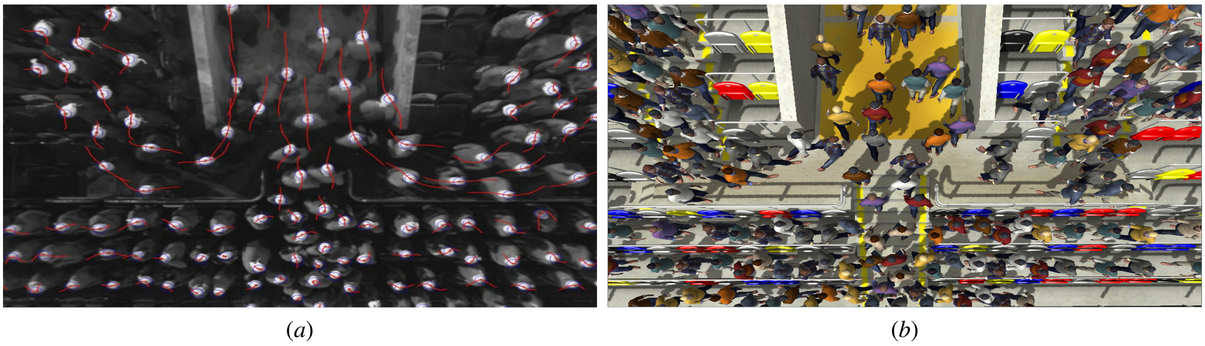

(*a*) (*b*)

Figure 1: *(a) A single video frame from a capture of a real world crowd exiting a stadium through a tunnel. The converging flows constrain the motion of spectators and lead to regions of high density. (b) The pedestrian simulation generated using our algorithm, DenseSense; it maintains the speed/density relationships observed in the real world video.*

Abstract

We present a novel algorithm to model density-dependent behaviors in crowd simulation. Our approach aims to generate pedestrian trajectories that correspond to the speed/density relationships that are typically expressed using the Fundamental Diagram. The algorithm's formulation can be easily combined with well-known multi-agent simulation techniques that use social forces or reciprocal velocity obstacles for local navigation. Our approach results in better utilization of free space by the pedestrians and has a small computational overhead. We are able to generate human-like dense crowd behaviors in large indoor and outdoor environments; we validate our results by comparing them with captured crowd trajectories.

1. Introduction

The problem of simulating realistic movement and behavior of crowds is important in many applications, including computer animation, games, pedestrian dynamics, virtual reality, etc. Crowd simulation has been extensively studied and a variety of techniques and models have been proposed to generate plausible, human-like crowd behaviors. Many of these models are based on *multi-agent simulation*, in which motion trajectories are computed for each individual agent in a crowd. There exists a subset of multi-agent algorithms that decompose the trajectory computation problem into two phases: global planning and local navigation. The global planner computes a path through the environment towards the current goal position of each agent while avoiding collisions with the static obstacles. The local navigation techniques take into account the path computed via the global planner and locally modify the trajectory to avoid collisions with dynamic obstacles or other agents in the environment. Collectively, this class of simulators will be referred to as

Global-Local Planners (GLP) and such combinations are frequently used [HFV00, CSS10, PAB07, Rey99, OPOD10, VGLM09, POO*09].

A key issue in these simulations of pedestrian movement is crowd density, which is typically measured in terms of average number of agents per square meter. Many studies in psychology, biomechanics, pedestrian dynamics and traffic management have indicated that pedestrian movement is sensitive to crowd density [IRTL81, Dea65, GLRM05, SSKB05, ZKSS11, ZKSS12]; as the density increases, the travel speed of each agent tends to decrease. This phenomenon is called the *Fundamental Diagram* [Wei93].

In this paper, we address the problem of modeling these crowd behaviors governed by crowd density: which we term *density-dependent behaviors*. Although human crowds do slow down in the presence of high densities, some of the well known local navigation algorithms based on social forces and reciprocal velocity obstacles do not exhibit such behaviors.

Main Results

We present DenseSense, a novel approach for density modeling that can generate realistic density-dependent behaviors and result in good space utilization. Our approach offers the following benefits:

- The speed and density relationship in the crowd movements computed by our algorithm adhere to the Fundamental Diagram.
- Our approach is general and doesn't make any assumptions about any global planning algorithm or local navigation technique.
- The computational overhead of our density-dependent filter is low and we can simulate hundreds of agents at interactive rates on a single core.
- The density-dependent filters result in smoother trajectories and also reduce the number of collisions between the agents.

The rest of the paper is organized as follows: We give a brief overview of the state of the art in Section 2. In Section 3, we present the details of our density dependent filter and the overall crowd simulation algorithm. We highlight its performance in Section 4. Finally, in Section 5 we analyze the model's strengths and weaknesses.

2. Related Work

In this section, we give a brief overview of prior work in crowd simulation and global and local planning.

Global trajectory planning includes a number of methods including navigation meshes, roadmaps and potential fields [LaV06], etc. Some work has been done to adapt these structures to dynamic obstacles [SGA*07]. Other techniques instead model congestion at the global level and compute paths that simply avoid these dense regions [GCC*10, KHR*11, vTCG12].

Local navigation methods include force-based [HFV00,CSS10,PAB07], rules-based [Rey99], vision-based [OPOD10], and velocity-space-based [VGLM09, POO*09] techniques. Recent models handle congestion by using global data structures for path computation [CM12] or physical forces [KGM13].

Density and crowd behaviors are closely related. Some prior crowd simulation algorithms sought to adhere to the Fundamental Diagram [LJK*12, HvdB13, GNL13]. Other models exist that do not fit the Global-Local class paradigm. One such model is the continuum model [NGCL09,TCP06], which treats humans as samples of a continuous medium. The notion of crowd densities and pedestrian motion has also been studied in other disciplines. Biomechanists have studied both personal space and the human gait at an individual level [IRTL81,Dea65]. There is extensive work in pedestrian dynamics on exploring the relationship between speed and density [SSKB05,ZKSS11,ZKSS12,CM12]. Our algorithm builds on many of these ideas.

3. Density-dependent Behavior

In this section, we present density-dependent filters and combine them with well-known local navigation algorithms. First, we introduce some of the notation and terminology used in the rest of the paper. In our crowd simulation algorithm, agents are modeled as two-dimensional disks with the following common *state*: $[\ r\ \mathbf{p}\ \mathbf{v^c}\ \mathbf{v}^0]^T \in \mathbb{R}^7$, where r is the radius of the disk, $\mathbf{p}$, $\mathbf{v^c}$, and $\mathbf{v}^0$ are two-dimensional vectors representing the current position, current velocity and the input velocity generated by the global planner, respectively. By convention v^c and v^0 are the current speed and the input speed generated by the global planner. We use subscripts to denote a particular agent i, such as r_i and $\mathbf{v}_i^0$.

3.1. Fundamental Diagram

The Fundamental Diagram is the observed relationship between pedestrian speed and density; as density increases, speed decreases. Many prior crowd simulation algorithms tend to reduce the speed of the agents through bottlenecks [GCC*10, HFV00]. However, this is largely a function of the reduced flow capacity of the *environment*, rather than the Fundamental Diagram. The Fundamental Diagram does not depend on specific environmental configurations; it corresponds to a relationship between the density and speed.

3.1.1. Physiological and Psychological Components

Our model incorporates two factors: a physiological and a psychological component. The physiological component is based on the biomechanical principle that stride length and walking speed are inherently linked [IRTL81]. The relationship [Dea65] is defined by a "stride factor" parameter (α). It relates a person's stride length (L) and walking speed (v) as:

$$L(v) = \frac{H}{\alpha}\sqrt{v}, \tag{1}$$

where $H = height/1.72\,\mathrm{m}$ is a height-normalizing constant.

The psychological component models the complex concept of personal space. Our model includes a “stride buffer” parameter (β) that reflects a preference for personal space [GLRM05]. As the stride buffer increases, an agent requires more space to walk comfortably at a given speed. We use a linear combination of the physiological and psychological components to yield a function that determines the natural walking speed (V) for the available space (S) in front of an agent:

$$V(S) = \max\left(v^0, \left(\frac{S\alpha}{H(1+\beta)}\right)^2\right), \tag{2}$$

where α is the stride factor as defined by [Dea65] and β is the stride buffer.

Space-estimation: Seyfried et al. [SSKB05] performed one-dimensional experiments on the Fundamental Diagram and observed that the 1D density was related to the 2D expectations defined by Weidmann [Wei93] by a scale factor equal to the inverse of the individual's width. We employ this observation in our model of space estimation by computing the local 2D density and then using Seyfried et al.'s method to map it to the scalar "space" value used in (2).

3.2. DenseSense

We introduce a density-dependent filter, DenseSense, that can be used to produce human-like density responses. DenseSense can be integrated with any Global-Local Planner (GLP) based model.

GLP based models use a global planner to plan a path through the static environment. The path is used to generate immediate goals which are communicated to the local planner as *preferred velocities* - velocities in the direction of an immediate goal, at a user-defined "preferred" speed. The local navigation scheme selects a viable velocity based on the preferred velocity and the local dynamic conditions (i.e. nearby obstacles and agents). DenseSense acts as an interface between the global planner and and the local navigation module, as shown in Figure 2. It seeks to reduce the mismatch between the global planner and the local planner by seamlessly adapting the preferred velocities generated by the global planner to local dynamic conditions. This results in smoother trajectories, reduced agent-agent collisions and realistic agent behavior.

At each step, our algorithm computes an arc centered at the input velocity from the global planner. It chooses a velocity from this arc that minimizes the cost function (9) while respecting the FD constraints. Finally, the density filter communicates this velocity to the local planner as the new preferred velocity for collision avoidance.

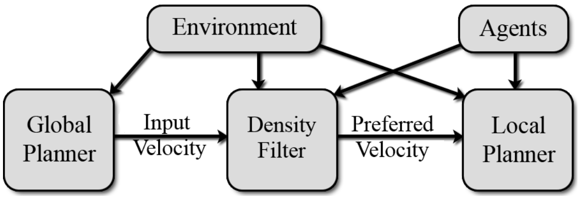


Figure 2: *DenseSense acts as an interface between the global and local planner.*

3.2.1. Agent Density Computation

Given an agent i and unit vector $\hat{\mathbf{v}}_i^\theta$ offset from the agent's input velocity $\mathbf{v}_i^0$ by an angle θ, we define a point $\mathbf{q}_i^\theta$ that lies one meter ahead along $\hat{\mathbf{v}}_i^\theta$. We use a Gaussian density function to determine the contribution of each neighboring agent to the density at $\mathbf{q}_i^\theta$. It is well known that humans exhibit an elliptical personal space, with a preference for space in front [GLRM05]. To model the personal space of agent i, we transform the relative positions of i's neighbors anisotropically, causing agents in front to have a greater contribution to density than those to the side (shown in (5) and (6)). Thus, the density of neighboring agents at $\mathbf{q}_i^\theta$ is:

$$\begin{aligned}
\rho_{Ai}^\theta &= \sum_j \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{\|\mathbf{d}'_{ij_\theta}\|^2}{2\sigma^2}} \qquad j \neq i, && (3)\\
\mathbf{q}_i^\theta &= \mathbf{p}_i + \hat{\mathbf{v}}_i^\theta, && (4)\\
\mathbf{d}'_{ij_\theta} &= 2.5(\mathbf{d}_{ij} - \mathbf{d}^y_{ij_\theta}) + \mathbf{d}^y_{ij_\theta}, && (5)\\
\mathbf{d}^y_{ij_\theta} &= (\mathbf{d}_{ij} \cdot \hat{\mathbf{v}}_i^\theta)\hat{\mathbf{v}}_i^\theta. && (6)
\end{aligned}$$

3.2.2. Effect of Obstacles on Density Computation

The presence of obstacles restricts the agent's available space and effectively increases the local density. We compute this impact by defining a local region (Ω), centered on the agent, and determine the *weighted* percentage of that region accessible to the agent. We consider regions close to the agent more important than those at the far boundaries of the domain. We compute the free space for an agent at $\vec{q}$ by solving the integral:

$$FS_q = \int_\Omega w(\vec{x})C(\vec{x}), \tag{7}$$

where $C(\vec{x})$ is 1 if x is accessible by the agent and 0 if not.

The effective density used in DenseSense computation for a point $\mathbf{q}_i^\theta$ is the agent density ρ_{Ai}^θ scaled by the inverse of the free space available at $\mathbf{q}_i^\theta$.

$$\rho_i^\theta = \frac{1}{FS(\mathbf{q}_i^\theta)}\rho_{Ai}^\theta \tag{8}$$

3.2.3. Agent's Response to Density

An agent i with width w_i chooses θ such that it minimizes the distance to the goal $\mathbf{g}_i$ with respect to time period τ

$$\arg\min_{\theta} \| g_i - (p_i + v_i^{FD_\theta})\tau \|, \tag{9}$$

where $\mathbf{v}_i^{FD_\theta}$ represents the Fundamental Diagram adherent velocity which can be computed as:

$$\mathbf{v}_i^{FD_\theta} = \frac{\mathbf{v}_i^\theta}{\|\mathbf{v}_i^\theta\|} V(\rho_i^\theta / w_i). \tag{10}$$

Finally, $\mathbf{v}_i^{FD_\theta}$ is provided as the preferred velocity to the local planner. We refer the reader to [NBC*14] for illustrations and expansion.

3.3. DenseSense Augmented Models

Our use of density-dependent filters is orthogonal to the choice of global and local planners. To illustrate this generality, as well as DenseSense's efficacy, we have applied it to several local planners. The simulators all share the same global planner, which is a navigation mesh. In particular, we demonstrate how our approach can be combined with two widely used local navigation algorithms: social forces (SF) [HFV00] and optimal reciprocal collision avoidance (ORCA) [VGLM09]. Henceforth, we will refer to the DenseSense augmented models as D-SF and D-ORCA respectively.

ORCA: Optimal Reciprocal Collision Avoidance (ORCA) applies geometric optimization techniques in velocity space. Avoiding collision simply requires selecting a velocity which does not lie within the reciprocal velocity obstacle set for each agent. ORCA's [VGLM09] unique formulation defines the velocity obstacles as half planes. The set of velocities that are permitted for an agent with respect to all other agents is the intersection of the half planes of permitted velocities induced by each other agent.

Social Forces (SF): The social force (SF) paradigm treats the crowd as a collection of mass particles. Newtonian-like physics is applied to the system to evolve the trajectories of the agents. At each time step, the superposition of various forces are computed for each agent, ultimately determining a viable velocity by imparting acceleration on the agent.

4. Results

In this section, we highlight the performance of our algorithm in different scenarios. Furthermore, we validate the results by comparing the simulated results with real-world data. In addition to generating density-dependent behaviors, our formulation also results in smoother trajectories and results in fewer agent-agent collisions as compared to prior methods. We base our implementation on the open-source, crowd-simulation framework, Menge [CBM14]. Menge provides a framework for global-local planning crowd simulators into which we inserted the DenseSense filter.

4.1. Validation

We validate the performance of our model in three reproductions of live pedestrian experiments. For each experiment, we report adherence to the Fundamental Diagram and indicators of trajectory smoothness based on SF, D-SF, ORCA and D-ORCA.

4.1.1. Two Dimensional, Uni-Directional Flow

Zhang et al. [ZKSS11] performed experiments of uni-directional flow in a corridor. The authors performed multiple iterations of the experiment, varying the flow into the corridor and the flow out of the corridor to control the observed density in the measurement region inside the corridor. Figure 3(a) shows the Fundamental Diagram results for ORCA, D-ORCA, SF and D-SF. Its easy to see that both ORCA and SF marginally reduce speeds at high densities. This is due to the constraints on the flow out of the corridor. However, with DenseSense, agents reduce speeds due to increase in density.

4.1.2. Two-dimensional, Bi-directional Flow

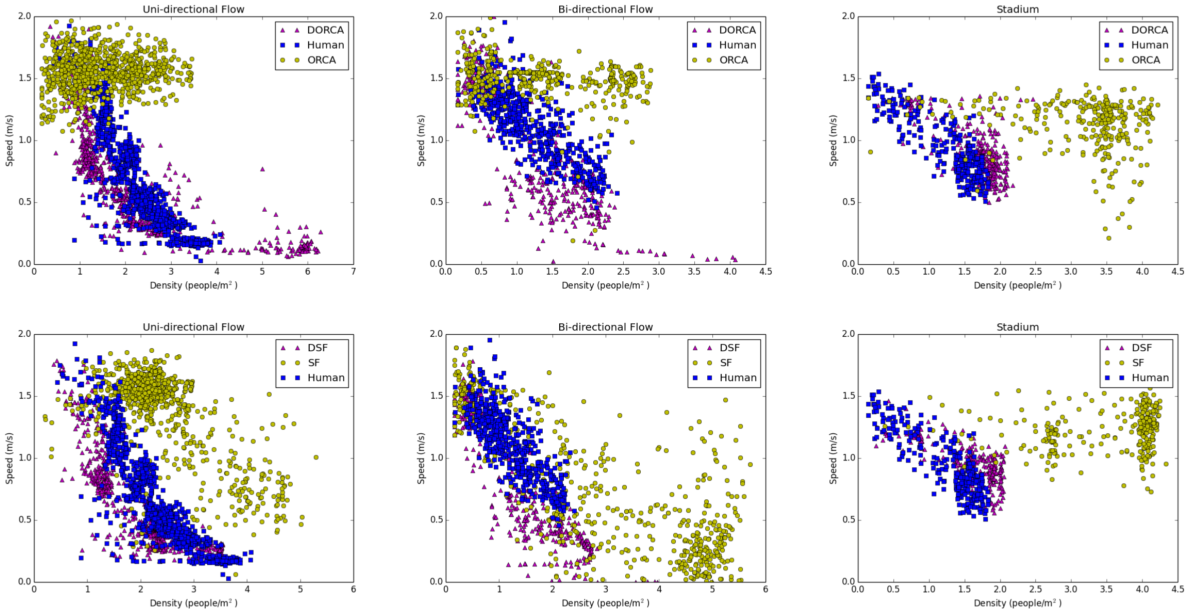
Following the uni-directional flow experiment, Zhang et al. also examined bi-directional flow [ZKSS12]. The Fundamental Diagram results can be seen in Figure 3(*b*). SF exhibited a reduction in speed, but with a profile inconsistent of that observed with pedestrians. In contrast, ORCA exhibited relatively constant speeds irrespective of density. However, with the application of DenseSense, both models exhibited density sensitivity consistent with that observed in the pedestrian data.

4.1.3. Stadium

Seyfried et. al. [BKS12] performed experiments on the complex flow of a crowd exiting a soccer stadium. They recorded the trajectories of 300 spectators as they made their way toward a predetermined exit tunnel. Figure 3(*c*) depicts the Fundamental Diagram, computed at the entrance of the tunnel, for this dataset. It is easy to see the success of DenseSense at effectively modeling the distinctive flow patterns in this complex scenario. In contrast, both ORCA and SF agents achieve significantly high densities and exhibit little or no response to density.

4.2. Applications

Aside from modeling real world data, we also tested the capabilities of our model on a number of additional scenarios, see [NBC*14] for more details. In each scenario, agents under D-SF and D-ORCA exhibited smoother trajectories and fewer collisions than their SF and ORCA counterparts.



(*a*) (*b*) (*c*)

Figure 3: *Fundamental diagram for SF, D-SF, ORCA, D-ORCA for the three experiments (a) The uni-directional 2D flow experiment in [ZKSS11] (b) The bi-directional flow experiment in [ZKSS12] and (c) The stadium experiment [BKS12]. Agents in both D-ORCA and D-SF exhibit a human-like sensitivity to high densities and conform to the Fundamental Diagram.*

4.3. Trajectory Smoothness

We illustrate our model's impact on trajectory smoothness by plotting trajectories of a small sample of agents (see Figure 4). We performed a baseline simulation using SF with a time step of 0.0625 s (middle). In addition, we simulated D-SF at the same time step (left) and SF at a 0.0001 s time step (right). The baseline trajectories are chaotic. Reducing the time step by a factor of 62.5 improved smoothness, but introducing DenseSense produced the smoothest trajectories, even at the larger time step.

5. Conclusion

We have introduced a novel crowd simulation algorithm based on density-dependent filters to generate human-like crowd flows. Our approach is applicable to a large number of GLP multi-agent algorithms that use a combination of local and global planners, and can generate realistic density-dependent behaviors. The computational overhead of DenseSense is relatively small, and the resulting approach generates smoother trajectories with fewer collisions [NBC*14]. Furthermore, the introduction of DenseSense requires no changes to either the global or local navigation components.

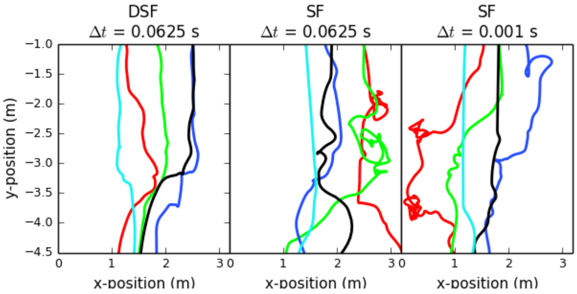


Figure 4: *DenseSense's impact on trajectory smoothness. Trajectories of five, sampled SF agents under three simulation paradigms: with D-SF at 16 Hz (left), SF at 16 Hz (middle), and SF at 1000 Hz (right). D-SF exhibited the smoothest trajectories.*

5.1. Limitations and Future Work

The potential impact of DenseSense depends on how sensitive the local planner is to the preferred velocity. The benefit of DenseSense may be diminished in scenarios where global density-dependent navigation techniques are already used. In future work, we will investigate the application of DenseSense at the meso-scale so as to allow agents to account for sudden changes in density. Furthermore, we plan to augment DenseSense with other psychological factors which may play a vital role in modeling dense heterogeneous crowds.

5.2. Acknowledgments

This work was supported by NSF awards 1000579, 1117127, 1305286, Intel, and a grant from the Boeing Company. We would like to thank the Julich Supercomputing Institute for their invaluable help in graciously providing their collected data and supplying a definition for the simulation domain. We would also like to thank Benjamin Chrétien for help with implementation.

References

[BKS12] Burghardt S., Klingsch W., Seyfried A.: Analysis of flow-influencing factors in mouths of grandstands. In *Pedestrian and Evacuation Dynamics PED* (2012). 4, 5

[CBM14] Curtis S., Best A., Manocha D.: *Menge: A Modular Framework for Simulating Crowd Movement*. Tech. rep., Department of Computer Science, University of North Carolina at Chapel Hill, http://gamma.cs.unc.edu/menge/, 2014. 4

[CM12] Curtis S., Manocha D.: Pedestrian simulation using geometric reasoning in velocity space. In *Pedestrian and Evacuation Dynamics* (2012). 2

[CSS10] Chraibi M., Seyfried A., Schadschneider A.: Generalized centrifugal-force model for pedestrian dynamics. *Phys. Rev. E 82*, 4 (2010), 046111. 2

[Dea65] Dean G. A.: An analysis of the energy expenditure in level and grade walking. *Ergonomics 8*, 1 (1965), 31–47. 2, 3

[GCC*10] Guy S. J., Chhugani J., Curtis S., Lin M. C., Dubey P., Manocha D.: Pledestrians: A least-effort approach to crowd simulation. In *Symposium on Computer Animation* (2010), ACM. 2

[GLRM05] Gérin-Lajoie M., Richards C. L., McFadyen B. J.: The negotiation of stationary and moving obstructions during walking: Anticipatory locomotor adaptations and preservation of personal space. *Motor Control 9* (2005), 242–269. 2, 3

[GNL13] Golas A., Narain R., Lin M. C.: Hybrid long-range collision avoidance for crowd simulation. In *ACM Symposium on Interactive 3D Graphics* (2013), pp. 29–36. 2

[HFV00] Helbing D., Farkas I., Vicsek T.: Simulating dynamical features of escape panic. *Nature 407* (2000), 487–490. 2, 4

[HvdB13] He L., van den Berg J.: Meso-scale planning for multi-agent navigation. In *ICRA* (2013), pp. 2839–2844. 2

[IRTL81] Inman V. T., Ralston H. J., Todd F., Lieberman J. C.: *Human Walking*. Williams & Wilkins, 1981. 2

[KGM13] Kim S., Guy S. J., Manocha D.: Velocity-based modeling of physical interactions in multi-agent simulations. In *Symposium on Computer Animation* (2013), pp. 125–133. 2

[KHR*11] Kretz T., Hengst S., Roca V., Pérez Arias A., Friedberger S., Hanebeck U.: Calibrating Dynamic Pedestrian Route Choice with an Extended Range Telepresence System. In *2011 IEEE International Conference on Computer Vision Workshops* (2011), pp. 166–172. 2

[LaV06] LaValle S.: *Planning Algorithms*. Cambridge, Cambridge, 2006. 2

[LJK*12] Lemercier S., Jelic A., Kulpa R., Hua J., Fehrenbach J., Degond P., Appert-Rolland C., Donikian S., Pettré J.: Realistic following behaviors for crowd simulation. *Comput. Graph. Forum 31*, 2 (2012), 489–498. 2

[NBC*14] Narang S., Best A., Curtis S., Chrétien B., Manocha D.: *Fundamental Diagram Adherent Interactive Crowd Simulation using Density-Dependent Filters*. Tech. rep., Department of Computer Science, University of North Carolina at Chapel Hill, http://gamma.cs.unc.edu/DenseSense/, 2014. 4, 5

[NGCL09] Narain R., Golas A., Curtis S., Lin M. C.: Aggregate dynamics for dense crowd simulation. *ACM Trans. Graph. 28* (2009), 122:1–122:8. 2

[OPOD10] Ondřej J., Pettré J., Olivier A.-H., Donikian S.: A synthetic-vision based steering approach for crowd simulation. In *Proc. SIGGRAPH* (2010), pp. 123:1–123:9. 2

[PAB07] Pelechano N., Allbeck J. M., Badler N. I.: Controlling individual agents in high-density crowd simulation. In *SCA '07* (2007), pp. 99–108. 2

[POO*09] Pettré J., Ondřej J., Olivier A.-H., Cretual A., Donikian S.: Experiment-based modeling, simulation and validation of interactions between virtual walkers. In *SCA '09* (2009), ACM, pp. 189–198. 2

[Rey99] Reynolds C. W.: Steering behaviors for autonomous characters. *Game Developers Conference* (1999). 2

[SGA*07] Sud A., Gayle R., Andersen E., Guy S., Lin M., Manocha D.: Real-time navigation of independent agents using adaptive roadmaps. In *Proc. ACM Symp. Virtual Real. Softw. Tech.* (2007), pp. 99–106. 2

[SSKB05] Seyfried A., Steffen B., Klingsch W., Boltes M.: The fundamental diagram of pedestrian movement revisited. *J. Stat. Mech.*, 10 (October 2005). 2, 3

[TCP06] Treuille A., Cooper S., Popović Z.: Continuum crowds. In *ACM SIGGRAPH 2006* (2006), ACM, pp. 1160–1168. 2

[VGLM09] Van den Berg J., Guy S. J., Lin M., Manocha D.: Reciprocal n-body collision avoidance. In *Inter. Symp. on Robotics Research* (2009). 2, 4

[vTCG12] van Toll W. G., Cook A. F., Geraerts R.: Real-time density-based crowd simulation. *Computer Animation and Virtual Worlds 23*, 1 (2012), 59–69. 2

[Wei93] Weidmann U.: *Transporttechnik der Fussgaenger*. Schriftenreihe des IVT, 1993. 2, 3

[ZKSS11] Zhang J., Klingsch W., Schadschneider A., Seyfried A.: Transitions in pedestrian fundamental diagrams of straight corridors and t-junctions. *J. Stat. Mech. 2011*, 06 (2011), P06004. 2, 4, 5

[ZKSS12] Zhang J., Klingsch W., Schadschneider A., Seyfried A.: Ordering in bidirectional pedestrian flows and its influence on the fundamental diagram. *J. Stat. Mech.*, 02 (2012), P02002. 2, 4, 5