Technische Universität München
Fakultät für Informatik
Lehrstuhl für Wissenschaftliches Rechnen

Simulating pedestrian dynamics: Towards natural locomotion and psychological decision making

Michael J. Seitz

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Alfons Kemper, Ph.D.
Prüfer der Dissertation:	1.	Univ.-Prof. Dr. Hans-Joachim Bungartz
	2.	Prof. Dr. Gerta Köster (Hochschule München)

Die Dissertation wurde am 23.03.2016 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 18.05.2016 angenommen.

Abstract

Social interactions and collective dynamics are ubiquitous in our lives and are increasingly being studied in computational science and engineering. Pedestrian dynamics focuses on the interactions and movements of humans on foot in a great variety of contexts, including public transportation systems and mass events. Computer simulation can serve both for studying pedestrian behaviour and as part of an information system – the former focusing on scientific scrutiny and the latter on engineering.

Due to the complexity of human behaviour, building a scientifically credible computer simulation is a great challenge for mathematical modelling and software engineering. Known approaches mostly focus on selected phenomena and do not incorporate findings from other disciplines, such as crowd psychology and biomechanics. Perhaps because of this, they lack a plausible representation of the decision making and physical process.

At first, I present a software framework that facilitates the development of new simulation approaches for pedestrian dynamics. I then discuss known approaches, and introduce the optimal steps model, which represents decision making through utility optimisation and locomotion as a discrete stepping process. The approach overcomes the limitations of the grid in cellular automata and is the first model of pedestrian dynamics representing the natural stepping process. However, according to findings in behavioural sciences, humans make decisions with heuristic reasoning, not mathematical optimisation. Therefore, I use a representation in separate layers for physics, psychology, and social behaviour in the second part of this work and propose dedicated simulation models drawing on findings from the respective fields. This approach constitutes a shift towards psychological and physical models that not only predict pedestrian behaviour but also serve as an explanation of the underlying processes. Specifically, I demonstrate how simple heuristics can be used to reproduce crowd phenomena and allow for new insights into pedestrian behaviour. The new concept facilitates interdisciplinary exchange and provides a basis for research in many directions, including psychology and biology.

Zusammenfassung

Soziale Interaktionen und kollektive Dynamiken sind allgegenwärtig in unserem Leben und werden zunehmend in Simulationswissenschaften untersucht. Fußgängerdynamik beschreibt die Interaktionen und Bewegungen von Menschen, die zu Fuß gehen, in einer Fülle von Kontexten, einschließlich öffentlicher Verkehrssysteme und Massenveranstaltungen. Computersimulationen können zur Untersuchung von Fußgängerverhalten oder als Teil eines Informationssystems dienen, wobei Ersteres auf naturwissenschaftlichen und Letzteres auf ingenieurswissenschaftlichen Erkenntnisgewinn abzielt.

Auf Grund der Komplexität menschlichen Verhaltens ist es für die mathematische Modellbildung und Software-Entwicklung eine große Herausforderung, eine wissenschaftlich glaubwürdige Computersimulation zu entwickeln. Bekannte Ansätze konzentrieren sich zumeist auf ausgewählte Phänomene und vernachlässigen die Forschung auf anderen Gebieten, wie etwa der Sozialpsychologie oder Biomechanik. Eventuell fehlt ihnen deswegen eine plausible Repräsentation des Entscheidungsverhaltens und der physikalischen Prozesse von Fußgängern.

Zuerst wird in dieser Arbeit ein Software-Framework vorgestellt, das die Entwicklung von neuen Simulationsansätzen in der Fußgängerdynamik ermöglicht und unterstützt. Danach werden bekannte Ansätze diskutiert und das Optimal Steps Model eingeführt. Das Optimal Steps Model repräsentiert die Entscheidungsfindung durch Nutzenoptimierung und die Fortbewegung als diskreten Schrittprozess. Der Ansatz beseitigt die Beschränkungen des Gitters in zellulären Automaten und ist das erste Modell für Fußgängerdynamiken, das den natürlichen Schrittprozess von Fußgängern abbildet. Der Forschung in Verhaltenswissenschaften zufolge treffen Menschen Entscheidungen mit Heuristiken und nicht durch mathematische Optimierung. Daher wird im zweiten Teil dieser Arbeit eine Darstellung in drei verschiedenen Schichten verwendet, jeweils eine für Physik, Psychologie und soziales Verhalten. Zudem werden dedizierte Modelle, die auf Forschungsergebnissen der jeweiligen Gebiete aufbauen, vorgeschlagen. Dieser Ansatz stellt einen grundlegenden Wechsel in Richtung psychologischer und physikalischer Modelle dar, die nicht nur das Verhalten von Fußgängern sondern auch der zugrundeliegenden Prozesse erklären. Insbesondere wird gezeigt, wie einfache Heuristiken verwendet werden können, um bekannte Phänomene zu reproduzieren und neue Einsichten in Fußgängerverhalten zu gewinnen. Die Darstellung in verschiedene Schichten fördert den interdisziplinären Austausch und bietet eine Grundlage für neue Forschung in viele weitere Richtungen, unter anderem in der Psychologie und der Biologie.

Contents

Chapter 1

Introduction

Social interactions are ubiquitous in our lives. Humans and animals show a great variety of social interactions in different contexts and to different ends. Especially as pedestrians, we often face others in daily routines such as commuting or when crossing the street but also during events, including music festivals with a large number of people. We interact with others by evading them, staying away with a certain distance, or staying close to chat with our friends while walking.

Although behaviours vary across animal species, certain similarities can be found. In particular, locomotion and navigation in the environment are common tasks for animals (e.g., McNeill Alexander, 2003). In the case of humans, the most natural form of locomotion is walking. As pedestrians, we cover many distances every day. Walking can have a recreational aspect or the specific purpose of reaching a destination. To give an example of the importance of walking: "Nearly 32% of all commuter trips in Delhi are walking trips" Tiwari (2003).

Pedestrian dynamics (e.g., Navin and Wheeler, 1969; Gipps and Marksjö, 1985; Helbing and Molnár, 1995; Schadschneider et al., 2009) mainly describes phenomena that result from the interaction of multiple pedestrians but may also study individual behaviour without social interactions. For example, the distance we keep to walls can be an important aspect in pedestrian dynamics. The study of crowd behaviour (e.g., Sime, 1995; Reicher, 1996; Faria et al., 2010; Drury and Stott, 2011) clearly stresses the social aspect of many humans interacting. Crowd behaviour can also be studied when humans are not walking or even are not physically in the same environment, which is the case in social media platforms. In this work, I study the behaviour of pedestrians who may or may not be in a crowd.

In computational science (e.g., Strang, 2007; Bader et al., 2013), natural phenomena are studied through computational methods. Mainly descriptive models are developed that explain known behaviour and generate additional hypotheses. These models may also be used for engineering problems and are of particular value for them if the models are well-validated, that is, produce reliable predictions. In computational engineering, descriptive *and* normative models are used. For example, in safety science, traffic optimisation, and the planning of built environments, a system that supports decision making is often desired. The motivations for the simulation of pedestrian dynamics are as plentiful as the disciplines involved. At first, studying pedestrian dynamics is an end in itself as basic research. Applications can be classified into improvements in safety, efficiency, and comfort for pedestrians.

A prominent example of safety applications is the study of mass events. A series of infamous crowd crushes (e.g., Elliott and Smith, 1993; Ahmed et al., 2006; Ngai et al., 2009) has drawn attention to the possible hazards in large agglomerations of people. The hope is to improve the safety of mass events with the help of simulations. For this, the layout of the environment, the number of visitors, and other parameters have to be known to produce meaningful predictions. Due to the complexity of human behaviour and the many parameters at such an event, a simulation model can reveal hazards but not exclude the possibility of additional, unknown ones.

The optimisation of technological systems is important economically, ecologically, and for safety reasons. At mass events, the efficient operation of services supports both safety and the satisfaction of visitors. In transportation systems (e.g., Vuchic, 2005), efficiency directly relates to the quality of the service from the customers' perspective. In these systems, passengers often move on foot a considerable portion of the time. Pedestrian simulations can be used for studies of railway stations or the design of trains (e.g., Köster et al., 2011a). Finally, well-designed public transportation systems promote their use through the comfort they provide.

Both the safety and comfort of pedestrians are issues in urban planning (Pucher and Dijkstra, 2000). Figure 1.1 illustrates how pedestrians in an urban setting often interact with each other and alternative modes of transportation. Pedestrian facilities are important for the attractiveness of cities, and walking has positive effects on social life and health (Leyden, 2003; Gehl, 2010). Transportation in big cities only seems to stay effective when functional public transportation systems are implemented and at the same time walking and cycling is promoted (Vuchic, 1999). There are approaches for the improvement of urban spaces to encourage people to walk (Southworth, 2005; Patton, 2007). Simulation studies may aid in design choices (Helbing et al., 2005) or even provide a virtual reality simulation as a visualisation tool for future developments.

Much of the work I present in this thesis was funded through the research project MultikOSi on assistance systems for urban events – multi-criteria integration for openness and safety. In the research group of Gerta Köster at the Munich University of Applied Sciences, we studied and developed simulation models of pedestrian dynamics. The main focus of the simulation approaches was on safety and efficiency issues in urban contexts. My work reflects this at some points, for example, with the study of shuttle-bus systems and of a railway station platform (in chapter 6). However, the discussions and simulation models are general and are not limited to these applications.

The simulation of pedestrian dynamics is a highly interdisciplinary field. Aspects of it are studied in many domains of the natural and life sciences, including physics, biomechanics, social psychology, and animal behaviour. The formal sciences and engineering contribute with mathematics, statistics, and computer science. Relevant disciplines in computer science are artificial intelligence, robotics, and scientific computing. Two main categories of research activities are empirical studies and model development. Both can stand on their own, but the combination of the two seems beneficial. Empirical studies without a theory that is being tested or developed out of them remain observations of singular events. Formal models that are not based on

Figure 1.1: Herald Square, 34th and Broadway, Manhattan. (Figure: From The New York Public Library, Abbott, 1935)

and tested with empirical observation remain theoretical.

Independent research in both categories is justified to a certain degree. Only from reliable experimental procedures and field observations can we deduce general statements. The formal models have to be consistent, translated into efficient algorithms that can be computed, and finally, they must be implemented correctly. Every one of these steps can be a great challenge. For a valid simulation model, the two directions of empirical and formal research must be combined. It is also crucial to study findings from related fields to complement the knowledge from the core discipline. Gigerenzer and Selten (2001, p. 10) write, “The lack of information flow between disciplines can hardly be underestimated.” Nevertheless, interdisciplinary research is also challenging because of the different methodology, terminology, and other factors. I discuss this issue in section 1.3.

1.1 Computational science and engineering

“A *model* is a representation of an event and/or thing that is real (a case study) or contrived (a use-case)” (Banks and Sokolowski, 2009). In this work, I focus on computational approaches for the study of pedestrian and crowd dynamics. Although interdisciplinary, the emphasis is on model development and simulation. Bungartz et al. (2014) give a general introduction to modelling and simulation. In this section, I discuss aspects of computational science and engineering that are relevant for the simulation of pedestrian dynamics. My intention is to give some background without the pretence of completeness.

It is difficult or not even possible to compare models and decide which one is better (Oberheim and Hoyningen-Huene, 2013). A model that may not yield the most exact prediction (e.g., Lattice Boltzmann and lattice gas models) may be preferred over a more precise model (the Navier-Stokes equations) because the latter can be computationally intractable. The quality of an approach is not only determined by the precision of the model but also by the usefulness, which can vary depending on the research question and application. Therefore, the objective of the undertaking has to be specified first. For example, scientific studies in psychology may have another focus than in biomechanics, and applications in safety engineering can have different requirements entirely, even if the research subject is the same. Some general dimensions for the quality of a model can be defined, which must be weighted according to the application.

For scientific progress, the falsifiability and testability of models are important (Popper, 2002), which is one reason why models must be parsimonious (section 1.3). Theories should be accurate and general at the same time. Neither a very general but imprecise model nor a detailed account of one specific event are useful theories. To allow for predictions, models must be theoretically and practically computable. A theory that does not allow for predictions because of its computational complexity cannot be tested. The model must be appropriate: it should predict what the user is interested in. Finally, the cost-benefit ratio has to be considered.

Models in science play an important role, but the meaning of the term is not that clear (Frigg and Hartmann, 2012). Since a model can also be a cardboard representation of a building in architecture, it is not necessarily a scientific theory. A model becomes a theory as soon as someone describes phenomena of reality with it. For example, a system of differential equations is only mathematical formalism, but as soon as someone claims it describes the movement of objects, it is a theory. In the philosophy of science, once a theory has been falsified, it is discarded (Popper, 2002). A model, on the other hand, may still be useful. A well-known example is Newtonian mechanics, which does not describe reality in all of its details and more precise models exist, but yet it is still useful.

To obtain meaningful simulation results, errors have to be excluded. “Sources for errors lurk in the model, in the algorithm, in the code or in the interpretation of the results” (Bungartz et al., 2014). Tests at different stages of the development process ensure that the model is correct. Two major categories of testing can be defined: verification and validation (Oberkampf and Roy, 2010). In short, verification ensures “solving the equations right” and validation “solving the right equations” (Strang, 2007, p. 714).

Verification makes sure that the model is computed correctly and should be undertaken at various stages. Given a formal model of equations, a numerical algorithm must be used to simulate the system in a computer. The algorithm is verified to guarantee that it produces results within a certain error range compared to the analytical solution of the equations. After the algorithm has been realised in software, the implementation has to be verified. Finally, the visualisation, statistical analysis, and other representations of the results must be verified, too.

Validation always compares model predictions with the real world. Validation relies on data obtained from empirical observations or controlled experiments. The empirical method can also be a source of errors. Especially in experiments with human subjects, the behaviour is easily influenced by the experimenter, the experimental design, or simply by the awareness of taking part in an experiment. These issues are addressed by a sound experimental design (e.g., Hinkelmann and Kempthorne, 2008).

In science, theories are tested by experiment in attempts to falsify them (Popper, 2002). If the theory does not withstand these attempts, it is discarded. In validation, models are compared to empirical data too, but the objective is to demonstrate the validity. Therefore, validation can be seen as the opposite of falsification, but a failed attempt to falsify a hypothesis is not automatically a good validation. A statistical test, such as the t-test, can only demonstrate that a hypothesis is wrong with a certain probability. The test does not provide the necessary information to show that the hypothesis is true and therefore is unreliable for validation. The unreflected application of statistical tests has also been criticised in general (Ioannidis, 2005). An alternative could be the application of equivalence tests (Robinson and Froese, 2004; Wellek, 2010). However, equivalence tests are not widely used in the scientific community and hence may not be accepted as evidence.

Empirical research serves both for the development and the validation of models. A sound experimental design is necessary to minimise errors (e.g., Hinkelmann and Kempthorne, 2008; Bailey, 2008). In computational science, experimental data is especially important to provide evidence for the validity of the model. In addition, the models sometimes have parameters that must be calibrated. The calibration of the parameters can be realised by using empirical data. In this case, a different data set is necessary for validation to prevent an overfitting effect (e.g., Hastie et al., 2009). The calibration of parameters can also be criticised in general: by simply recalibrating the model for every scenario one can evade falsification. Therefore, the parameters should be considered as part of the model. Every recalibration leads to a new model. If the parameters are not considered part of the model, model predictions have to be valid with any set of parameters or any parameter within a specified range of values. Mostly, this can only be shown analytically, and then the predictions tend to be unspecific.

1.2 Pedestrian dynamics

A series of phenomena is studied in pedestrian dynamics (e.g., Navin and Wheeler, 1969; Gipps and Marksjö, 1985; Helbing and Molnár, 1995; Schadschneider et al., 2009). They can describe both the behaviour of individuals and of any size of pedestrian agglomerations. In safety applications, mainly measurements of egress times and crowd densities are of interest. In studies on efficiency, mostly the time necessary to reach a certain state is studied. Examples are passenger exchange times or the time it takes to board an aeroplane.

Important phenomena are the lane formation in contra-flow scenarios (Helbing and Molnár, 1995; Kretz et al., 2006a) and the characteristic density-speed relation, often called fundamental diagram (Navin and Wheeler, 1969; Seyfried et al., 2005; Jelić et al., 2012a), which are both well-documented in controlled experiments and in simulation studies. Particularly relevant for evacuation scenarios is the faster-is-slower effect (Kelley et al., 1965; Helbing et al., 2000a; Garcimartín et al., 2014;

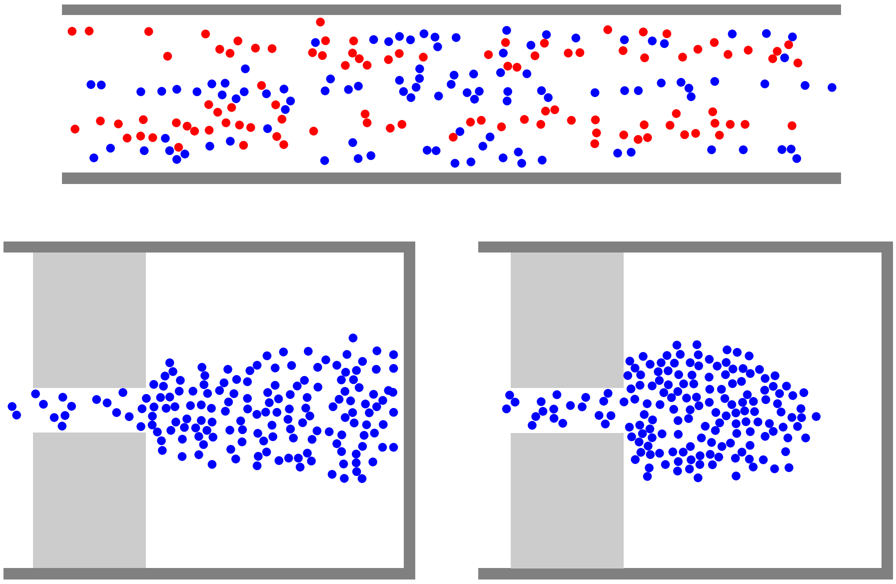

Figure 1.2: Simulation snapshots from typical scenarios. On the top, simulated pedestrians walk from left to right (red) and right to left (blue) and form lanes, which facilitates flow. On the bottom, simulated pedestrians egress from the room on the right through a corridor. A congestion forms in front of the corridor. This scenario is especially interesting since bottlenecks can be the decisive part of the environment that determines evacuation times. (Figure: Seitz et al., 2016)

Pastor et al., 2015), which suggests that increasing motivation to leave a room fast eventually leads to slower egress. This may be due to phenomena of arching and clogging at bottlenecks that are well-known in granular flow (Pöschel, 1994; To et al., 2001). Empirical evidence for this effect is sparse: it is difficult to conduct experiments with a very competitive behaviour of participants (Pastor et al., 2015). In figure 1.2, two typical scenarios, contra flow and egress through a bottleneck, are shown with data from a simulation.

Other phenomena include stop-and-go waves (Helbing et al., 2007), turbulences in very dense crowds (Helbing et al., 2007), oscillations at bottlenecks (Helbing and Molnár, 1995; Kretz et al., 2006b), and circulating flows at intersections (Helbing et al., 2005). A behaviour on a smaller scale is the coherence of social sub-groups (James, 1953; Coleman and James, 1961; Aveni, 1977; Singh et al., 2009; Moussaïd et al., 2010). On the individual level, effects such as the distances pedestrian keep to walls and each other as well as walking speeds can be of interest (Moussaïd et al., 2009b). It seems difficult to find a general law for pedestrian dynamics as even the walking speed is different between cities (Bornstein and Bornstein, 1976; Levine and Norenzayan, 1999). Considering this, it is not surprising that Chattaraj et al. (2009) found that the density-speed relation varies across cultures.

Guidelines for the validation of pedestrian stream simulations exist. RiMEA (Richtlinie für Mikroskopische Entfluchtungs-Analysen[1], RiMEA, 2009) is a German

[1]German, guidelines for microscopic evacuation analyses.

guideline maintained by an association of research institutes and organisations from the private sector. It provides basic tests for the verification and validation of evacuation simulations. Another guideline is the NIST Technical Note 1822 on "The Process of Verification and Validation of Building Fire Evacuation Models" (Ronchi et al., 2013), published by the National Institute of Standards and Technology (NIST) in the United States.

A variety of conceptual modularisations for models of pedestrian dynamics has been proposed. In artificial intelligence, the agent components are often divided into sensing, planning, and acting (Russell and Norvig, 2010). Another separation is the path-velocity decomposition from robotics in "(1) planning a path to avoid collision with static obstacles and (2) planning the velocity along the path to avoid collision with moving obstacles" (Kant and Zucker, 1986). Gipps and Marksjö (1985) separated the model in route choice and locomotion ("movement along a link"), which is not the locomotion as I understand it in this work but rather local navigation. Hoogendoorn and Bovy (2004) defined three levels: strategic (1), tactical (2), and operational (3). Each of the levels gathers certain functionality: "(1) activity choice behaviour and activity area choice, (2) wayfinding to reach activity areas and (3) walking behaviour" (Hoogendoorn and Bovy, 2003). However, it is not always clear what a strategic, tactical, or operational decision is. For example, local walking behaviour includes keeping a certain distance to obstacles, which is also a strategic decision. Wijermans (2011) proposed "three levels of description: the group level (inter-individual), the individual level and the cognitive level (intra-individual)". Reynolds (1999) used a separation into action selection, steering, and locomotion, and Hoogendoorn (2007) separated physical and control models.

1.3 Challenges in simulating pedestrian dynamics

The simulation of pedestrian dynamics poses an interdisciplinary challenge. Human behaviour is highly complex and varies depending on the individual and the context (e.g., Matsumoto, 2012). Behaviour can often only be predicted stochastically, that is, on average. Different scientific disciplines have different approaches to the study of behaviour: while psychologists also consider internal aspects such as motivation, biologists mostly have to rely on observed behaviour.

There are controlled experiments that can reproduce certain crowd phenomena and allow for measurements, including density and speed (Steffen and Seyfried, 2010). The methodology is rather quantitative, relying on the analysis of video footage. This approach mainly describes behaviour, and hence, the simulation models based on them can be considered phenomenological descriptions of pedestrian dynamics. The same is usually true for field observations. In psychology, on the other hand, models are often less formal and not in closed form. A different methodology is employed, such as interviews and questionnaires, which may reveal internal factors like motivation.

Mathematical modelling and simulation are in part formal sciences since they deal with equations and algorithms. This is also true for physics, which is a natural science but heavily makes use of mathematical formalism. In psychology, many theories are not formalised. Carrying them over to computer simulation may not seem possible at first. If that were truly the case, it would be a fundamental problem for the simulation of pedestrian dynamics because it describes human behaviour as does psychology.

Mathematical modelling as the basis of computer simulation relies on certain principles[2]. Perhaps most important are abstractions from the real world, also referred to as Aristotelian and Galilean *idealisations* (Frigg and Hartmann, 2012). Aristotelian idealisation means that features of reality are deliberately left out of the model. Galilean idealisations describe the deliberate distortion of the real world. Both can be thought of as simplifications or abstractions. An example of an Aristotelian idealisation in pedestrian dynamics is that we do not consider the hair colour of individuals because we expect it to have little impact on the phenomena we want to describe. The representation in the two-dimensional transverse plane could also be considered an Aristotelian idealisation as it omits a whole dimension. An example of a Galilean idealisation is the representation of pedestrians as circles, which clearly is a distortion of the real world.

The simplification through idealisations is the principal method to come up with models that describe phenomena formally and finally mathematically. For the simulation in a computer, the mathematical model has to be closed, and every parameter has to be fixed. In some cases, methods such as uncertainty quantification (e.g., Smith, 2014) may help to study a whole distribution of parameters, but the simulation itself is always run with one specific set of parameters. The formal description and finally the implementation in software allows for the study of emergent phenomena predicted by the model.

In computational science, the model encodes a theory that can now be tested and possibly be falsified if the prediction contradicts empirical observation. This is of utmost importance because falsification is the basic principle of the scientific method (Popper, 2002). With too many parameters in a model, it can be easy to evade falsification by simply adapting the parameter every time the prediction did not match empirical observation. Therefore, the model with its specified set of parameters has to be considered a theory that is being tested and potentially falsified.

A model with a set of possible parameters can still be considered a theory under certain circumstances: the prediction must be tested for all of them. Then statements of the form "theory X (the model with a specific parameter or a set of possible parameters) predicts P" can be formed and tested. In contrast, statements like "there is a parameter for theory X that predicts P" may also be interesting but do not represent scientific theories. Those statements do not match the structure of a scientific theory because they cannot be falsified unless all parameters are tested. In fact, if all parameters are tested or a parameter is found that verifies the statement, it is not a theory anymore but a singular statement, which is not a scientific theory by definition (Popper, 2002).

Whenever two models are known that both predict the same phenomenon, one has to be selected. In practice, the model that matches the purpose best is chosen. Then the criterion can be, for example, faster computation. In science, the more parsimonious model is preferable, although this criterion is not always clearly defined.

[2]The following ideas and the line of argument in this section were partly developed in collaborative work with researchers at the University of Sussex (Seitz et al., 2015d).

Popper (2002, sections 41–46) discusses this issue and proposes to use the "degree of falsifiability" instead of "simplicity" as a criterion: "The degree of universality and of precision of a theory increases with its degree of falsifiability" (Popper, 2002, p. 127). Thus, the requirement of parsimony or degree of falsifiability is important. It allows to discriminate among models and will ultimately lead to the most accurate and general theories.

Social psychologists (e.g., Drury and Stott, 2011), on the other hand, criticise that the approach mathematical modelling has taken for the explanation of crowd phenomena is reductionist. When comparing simulation outcomes to real events, they found that simulation models neglect crucial features of the crowd's behaviours. Social psychology deals with open systems that strongly depend on the context. This poses a challenge for mathematical modelling. However, the principles of the scientific process hold, and hence, there is no underlying contradiction between the requirement of parsimony and avoiding reductionism in explaining crowd phenomena. The challenge rather lies in formalising models from social psychology without distorting them too much and selecting scenarios that can be simulated in a meaningful way. When successful, mathematical modelling and computer simulation can also help to test theories from social psychology.

Although there may not be a fundamental contradiction, there are still problems on a more practical level. The scientific language and applied methodology are different and it takes time to overcome this barrier. Nevertheless, surprising similarities can be found and new aspects discovered. Others have already worked at closing the gaps between disciplines, and we can draw on their experience. We are still at an early stage when it comes to simulating human behaviour and the models we have are far from perfect. This should not prevent us from working on it, but the resulting models must be treated with the necessary care and should not be understood as definite representations of human behaviour.

Most simulation models of pedestrian dynamics do not make this step and remain merely phenomenological descriptions. It can be a limitation when the underlying processes are not studied. Moussaïd et al. (2011) and Moussaïd and Nelson (2014) argued for process-oriented simulation models in pedestrian dynamics. These models do not only describe phenomenologically but also try to represent the underlying process. In the case of pedestrian dynamics, this is mainly locomotion, decision-making, and collective behaviour. The hope is that this perspective allows for more accurate predictions as well as a better understanding of pedestrian behaviour.

In order to develop a simulation model that represents a plausible underlying process, it is necessary to refer to the disciplines that study them, which are physics, psychology, and biology, among others. And here the real challenge lies: it is hardly possible to integrate the whole body of knowledge in any one model. Models have to be kept simple for a variety of reasons, including computability. Therefore, it is essential to look at other disciplines but also to select focal aspects for the integration into a simulation approach. This is reflected in my thesis as I select important aspects from domains such as biomechanics, artificial intelligence, cognitive sciences, and social psychology without the claim of completeness.

1.4 Scope and overview of this work

In this thesis[3], I develop models that aim at a better understanding of pedestrian dynamics, especially the underlying processes. Given the many interactions in pedestrian crowds, a computer simulation is necessary for model predictions. The hope is that with understanding and representing the underlying mechanisms, I obtain reliable simulation results, better understand the described phenomena, and provide a suitable basis for future directions in model development. The motivation is mainly scientific scrutiny but also includes a series of applications, such as transportation research and safety engineering.

Computer simulations are developed in computational science and engineering (e.g., Banks and Sokolowski, 2009). In this field, computational methods are used to provide tools for studies in science and engineering. Therefore, the computational tools are a strong focus. In my work, I focus on model development, especially from a conceptual perspective. Other important areas in mathematics and computer science include scientific computing, analysis, statistics, and software engineering, which are not the focus of this work but are necessary for the study of simulation models.

When developing a simulation approach, it is important to choose the right scale that captures the features of interest. To better understand the underlying processes in pedestrian dynamics, I study individual-based approaches, often called microscopic simulations. Simulations that do not represent the individual, often called macroscopic simulations, are inapt for the representation of the underlying processes. They do not provide the necessary resolution and, for example, make it impossible to study phenomena such as the stepping behaviour or other biomechanical features of walking. Too detailed models, on the other hand, are also not a good choice. For example, a neuronal model of the brain is not suitable for pedestrian dynamics since the details obstruct the view on the mechanisms of interest. Nevertheless, some detailed approaches, for example, models from biomechanics or wider perspectives on a group-level, can complement the models I propose.

I focus on two-dimensional simulations in the transverse plane (top-down view). The transverse plane allows for the study of pedestrian motion without the overhead of the three-dimensional world we live in. This idealisation of the real world seems justified as pedestrians mainly move in the transverse plane. For biomechanical features, a three-dimensional representation could be of interest but is not developed in this work.

The simulated individuals – pedestrians in this case – are called *agents* throughout this work to clearly differentiate them from the real world. Agents are also a common concept in artificial intelligence that I refer to (in section 6.2, chapter 6). Some phenomena of group behaviour can be reproduced by the interactions of individuals without a group-layer perspective. Gipps and Marksjö (1985) formulated this direction:

[3] Whenever I use the personal pronoun *we* in the context of my work, I refer to myself together with colleagues, which means the specific work was either collaborative or has been published together. I created all figures used in this work with the exception of figures 1.1, 5.2, 5.4, 6.15, and 6.16. More detailed credit is given in the respective references cited in the text, captions, and footnotes.

> The simulation is tackled at the level of the individual pedestrian under the hypothesis that if the behaviour of individuals is modelled adequately, and the appropriate distribution of pedestrian types employed, the corporate behaviour of the simulated pedestrians will be realistic. Further, by working at the level of the individual it is possible to collect data on individual travel times and diversions, and subsequently to analyse the variability between different types of pedestrian. (Gipps and Marksjö, 1985)

There is some criticism concerning this rationale. Especially the reality of groups and their impact on the world should not be neglected:

> Cognitive scientists tend to focus on the behavior of single individuals thinking and perceiving on their own. However, interacting groups of people also create emergent organizations at a higher level than the individual. (Goldstone and Janssen, 2005)

My thesis is divided into two parts: part I describes approaches for the simulation of pedestrian dynamics from a classical perspective; part II is dedicated to a modelling approach towards a natural process-oriented representation of underlying mechanisms. Many of the tools and topics I discuss in the first part form the basis for the second part. At the same time, I point out the limitations of the models in part I in reference to part II.

In the next chapter, I start by describing a simulation framework that forms the technological basis for the implementation of models from pedestrian dynamics. The particular focus of this chapter is the modular design that promotes changes and new developments. In chapter 3, I review microscopic simulation models of pedestrian dynamics. I classify the models, identify commonalities and differences, and discuss their limitations. In chapter 4, I present the optimal steps model, an approach that captures the natural stepwise motion of pedestrians and uses it as discretisation scheme in the simulation. The optimal steps model overcomes limitations of cellular automata and represents an advancement towards a natural locomotion process. It relies on utility optimisation for decision making, a coherent concept that is accessible to a wide range of disciplines.

I use a separation of the underlying processes of pedestrian dynamics in part II. From an engineering perspective, the modularisation in artificial intelligence and robotics are reasonable. The strategic, tactical, and operational level used by Hoogendoorn and Bovy (2004) are useful if sub-models fit into the categories. However, in this work, I try to capture the underlying mechanisms. I choose a separation in layers that reflects the processes involved. Moussaïd et al. (2011) used a locomotion layer for the physical process and a decision-making layer for the psychological process. I follow this separation. Additionally, I introduce a social layer that describes models that build on the underlying two layers. This can be additional aspects of individual social interactions or – going beyond it – models of collective behaviour on the group level. The layers and their composition are illustrated in figure 1.3.

The chapters in part II map onto this separation: chapter 5 is dedicated to the physical layer, chapter 6 to the psychological layer, and chapter 7 to the social layer. At the beginning of each chapter in the second part, I show a figure with the layers

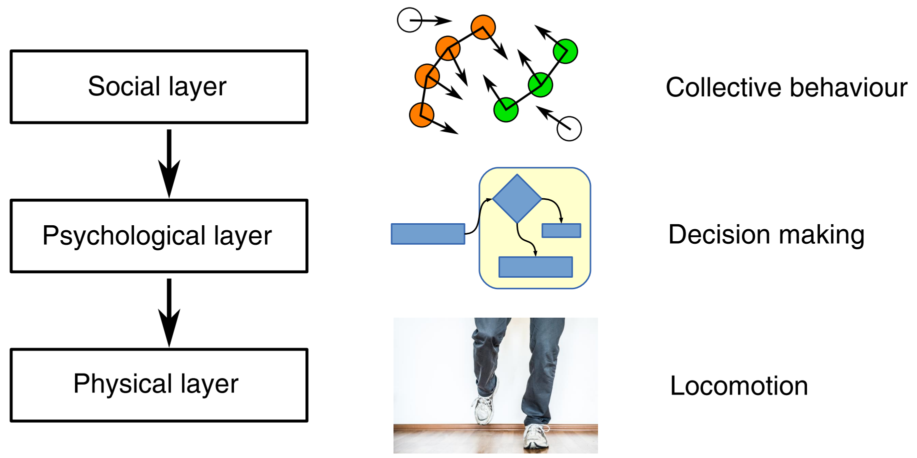

Figure 1.3: Illustration of the separation into three layers. The physical layer represents the locomotion of pedestrians and other physical aspects such as contact forces (chapter 5). The psychological layer describes individual decision making and builds on a physical layer (chapter 6). It also provides a basis for the social layer, which contains models of collective behaviour and social interactions (chapter 7). The arrows indicate that the layers on top depend on the ones below.

highlighting the layer the chapter is dedicated to. For the physical layer, I propose a discrete process for efficient computation and a force-based process for detailed, continuous simulations. The core concept on the psychological layer are cognitive heuristics (e.g., Gigerenzer and Todd, 1999). I present a model for pedestrian navigation in the proximity based on this paradigm. On the social layer, I present simulation approaches for sub-groups of up to four members and a conceptual treatment of how to introduce crowd models from social psychology, namely, the social identity approach (e.g., Reicher, 1996; Drury and Reicher, 1999, 2010), into computer simulation.

Part I

Microscopic pedestrian stream simulation

Chapter 2

Simulation framework

The software framework Vadere[1] (Latin, to walk) has been developed as a platform for researching pedestrian simulation models. Our main objective was to dispose of an independent framework that promotes research and provides a basis for educational purposes, especially seminars and theses. It had to facilitate model extensions, new model developments with as few constraints as possible, and studies comparing models. These objectives entail certain requirements, including tools for visualisation, data output, and collaborative work on code and models. In this chapter, I describe the requirements, the development process and toolchain we use, and the software design, and give reasons for the respective choices. After this preparation, I present the functionality and software architecture we implemented in Vadere and point out how it meets the requirements and design.

We need full access to the code in order to verify which algorithms are used and how they are implemented. The source code must also be accessible for educational reasons to instructors and students. We must be able to change and extend the software in all of its aspects. Existing software that is not published as open source is unfit for our purpose, and thus, only open source projects are considered in the following. We plan on publishing our own framework as open source project in the future and with this provide other research groups full access to the framework.

There are a number of open source software projects available for microscopic pedestrian and traffic simulation. Table 2.1 and 2.2 summarise them in their alphabetical order. All of them are available with their source code and hence are in principle suitable for our purpose. The main objective of MATSim (MATSim Contributors, 2015) and SUMO (SUMO Contributors, 2015) is traffic simulation. Implementations

[1]Vadere has been developed in the research group of Gerta Köster at the Munich University of Applied Sciences. I gave the principal direction for the requirements, development process, software design, and software architecture. Felix Dietrich and I implemented the core framework starting with legacy code by Swen Stemmer and myself. Felix Dietrich contributed to many of the results presented in this chapter. Specifically, he put forward the use of JSON as a markup language for the simulation parameters and implemented the respective routines, implemented core functionality and geometrical operations, and contributed to the design of the software architecture. Benedikt Zönnchen and Felix Dietrich gave the principal direction for the graphical user interface and the post-processing units and implemented the greatest portion of it. Isabella von Sivers contributed code to the implementation of the optimal steps model. Gerta Köster set the basic objective of providing a software framework to research modelling and simulation of human behaviour, in particular, pedestrian motion.

Name	Initiating institutions	Licence
FDS+Evac	VTT Finland	Public domain
JuPedSim	Jülich Forschungszentrum	LGPL
MATSim	ETH Zürich, TU Berlin, Senozon	GPLv2
Menge	UNC Chapel Hill	custom
PEDSIM	ETH Zürich, TU Berlin	GPLv3
SUMO	DLR Berlin	GPLv3

Table 2.1: Available open source simulation software that could be used as a basis for the simulation of pedestrian dynamics. The second and third column list the institutions that initiated the project and the licence information.

Name	Simulation model	Language	Reference
FDS+Evac	social force, FDS	Fortran	FDS Evac contributors (2015)
JuPedSim	force-based, routeing	C++	JuPedSim Contributors (2015)
MATSim	agent-based, queueing	Java	MATSim Contributors (2015)
Menge	multi-layered, generic	C++	Curtis et al. (2015)
PEDSIM	social force	C++	PEDSIM Contributors (2015)
SUMO	car following	C++	SUMO Contributors (2015)

Table 2.2: Open source simulation software with implemented simulation models and programming language used.

of the social force model can be found in PEDSIM (PEDSIM Contributors, 2015) and FDS+Evac (FDS Evac contributors, 2015). FDS itself is primarily a fire simulator that also features crowd simulation in the extension Evac. JuPedSim (JuPedSim Contributors, 2015) has two force-based models as concrete implementations so far and offers algorithms for the routeing of simulated pedestrians. Menge (Curtis et al., 2015) is designed to be generic and feature arbitrary models, but the software design imposes a certain model structure not all models fit into (Curtis et al., 2016).

When assessing the available frameworks, all of them show some features that could be built on. Only MATSim is written in Java but is rather dedicated to traffic simulations than to pedestrian dynamics. SUMO is also mainly a framework for traffic simulations. PEDSIM is not necessarily designed as a framework but rather a dedicated implementation of the social force model. JuPedSim and Menge are indeed designed as frameworks but were not available when we started the project.

In the following section (section 2.1), I develop and collect requirements. In a research project, there are particular requirements – not only for the code itself but also the development process and toolchain. I describe the process and toolchain we adopted in section 2.2 and discuss the software design and explain the choices made in section 2.3. Given this background, I outline the functionality (section 2.4) and software architecture (section 2.5). This may serve as a documentation but also suggest solutions for similar problems. Finally, I summarise the chapter and give an outlook on possible future developments in section 2.7.

2.1 Software requirements

According to the scope of this work (section 1.4 in the introduction), the simulation framework aims at studying microscopic simulations of pedestrian crowds. Therefore, it must facilitate the investigation of simulation models, including the comparison of models, the assessment of suitability for application, and the study of similarities and differences. It is necessary that the framework be suitable for the fast-changing requirements in a research project and hence allow flexible model development. For example, it should be easy to extend, exchange, or remove models from the software without changing the framework. Comparing models requires the possibility to maintain a variety of simulation models at the same time without them interfering with each other. These demands pose an immense practical challenge and are the main non-functional requirement. In fact, the non-functional requirements are as important as the functional ones in this situation. Balzert (2009) gives a general introduction to requirements engineering.

To illustrate the purpose of the software, I outline the functional requirements first. The overall function of the software is to run simulations of pedestrian crowd behaviour using a variety of models, including the social force model, cellular automata, and the optimal steps model. The parameters, which are the scenario, model parameters, and output parameters, have to be specified and stored in an accessible format so that the simulation can be replicated. The software has to produce output describing the trajectories of individuals in the simulation that can be used to visualise or analyse the simulation outcome. The framework has to offer online processing of the simulation such as density and speed measures for individual agents or within a measurement area. An online visualisation must be available to indicate the current state of the simulation, which also allows for immediate visual, qualitative verification and validation of simulation outcomes. A post-visualisation based on the simulation output is necessary and must allow for observing the simulation with different speeds and for jumping in simulation time. This also includes features such as taking snapshots and storing them in different formats or capturing videos of the running simulation. Finally, all features that require user interactions should be integrated into one graphical user interface (GUI).

The users of the simulation are university students and researchers. Almost every user is also a developer as studying simulation models mostly means to modify parts of it or, at least, understand its implementation. Therefore, only open source software can be used. This excludes any other framework with a proprietary licence and software packages that are not available as open source. Apart from the accessibility of open source software, it is also usually available free of charge and can be distributed (under the respective licence). This is especially important for educational purposes.

To allow developers to perform simulations independently and quickly assess the results of their implementation, the software has to run on desktop hardware. Making the software available for desktop hardware is also important as students are designated users of the software. Due to the variety of users and developers, the framework should be platform independent – especially run on Microsoft Windows, Linux desktop environments, and Mac OS. This also facilitates large-scale simulations on high-performance platforms in addition to small studies on desktop hardware.

Software requirements

Functional	- run simulations of pedestrian crowd behaviour - specify and store parameters in text files with a simple format - generate output that describes the trajectories of individuals - online processing of the simulation - online and post-visualisation - integrated graphical user interface
Non-functional	- only use open source software - run on modern desktop hardware - platform independence - object-oriented, high-level programming language - implement new models without changing the framework - framework must not impose any model concept or structure - modular design and architecture - reusability of basic algorithms and data structures

Table 2.3: Functional and non-functional software requirements.

The programming language used must be a widely known, object-oriented, and a high-level language that facilitates a modular and effective style. For the simulation framework Vadere, we decided on Java (Oracle, 2015a) for a variety of reasons. The compiler and runtime environment are available for many operating systems, especially Windows, Linux, and Mac OS. It is open source and has an active community (Cass, 2015). Its principle programming paradigm is object-oriented, but newer versions also allow for concepts known from functional programming such as lambda calculus. It is widely used (Cass, 2015), and good practice guidelines are available that facilitate collaborative software development (Vermeulen et al., 2000; Bloch, 2008).

Apart from the requirements on the basic software infrastructure, there are additional, more specific demands on the software design. The implementation of new models should be possible without changing the framework, and at the same time, the framework must not impose any modelling concept or structure that could hinder new model developments. It should be possible to combine models or parts of models – an objective that can be realised on demand. Basic functionality, including numerical algorithms, and geometrical data structures and routines, should be available to all models independently. Table 2.3 summarises the requirements.

General code and software requirements such as readability, conciseness, reusability, flexibility, genericity, and moderate implementation time are difficult to meet at the same time or may even contradict each other. Therefore, one has to fulfil the most important requirement and try to meet the rest as well as possible given the specific case. When there is a conflict of goals, it may even be necessary to ignore some requirements. The whole process can be considered a multi-objective optimisation: finding a suitable solution to a software problem is a Pareto optimisation (of the multiple requirements), and the objective is to find a solution in the Pareto optimal set. In practice, whether the requirements are met is not as clearly defined and often depends on individual judgement.

2.2 Development process and toolchain

The purpose of the simulation framework is to provide a platform for researching pedestrian simulation models. I derived requirements from this objective, which are taken into consideration in the software design (section 2.3). The focus on a research platform also has implications for the development process and toolchain. The following concepts are common in software engineering in general but seem especially important for the required flexibility in research.

We consider the framework as a tool to study models of pedestrian crowd behaviour. It is not a product or a prototype of a product for end-users. The users are rather students and researchers who study simulation models and hence are usually developers at the same time. Since we study simulation models, extensions and new models are introduced frequently. Therefore, the requirements on the platform can change too and are difficult to foresee. This led us to the conclusion that an agile software development process is suitable for our project.

With agile software development (Dingsøyr et al., 2010), responding to change is emphasised and valued more than the pre-planned software design. For example, new requirements can be implemented quickly to produce working functionality and then the software design is adapted accordingly. A change in design is usually carried out in a refactoring process rather than before new features are implemented, which is especially reasonable when requirements are not known before. Moreover, I tried not to impose patterns on parts of the software that are researched because we cannot assume that we know future developments. Imposing a structure may inhibit the flexibility in studying new concepts.

To support collaborative software development, we used the distributed version control system Git (Git Contributors, 2015). The remote repository is provided by a server. As the full history of the repository is always stored locally too, we implicitly maintain a distributed backup system. The revision number of the software is stored in the simulation output, which makes it easy to replicate simulation runs. For this, the current revision number is written into a file by a Git hook that is triggered after the current version has been checked out. The file with the revision number must not be in the repository because the version number is based on a hash of the current state of the repository. It either has to be placed outside of the repository folder or set to be ignored by Git. We chose the latter option. The framework reads the file and writes the revision number into the output document.

For programming, I used the integrated development environment (IDE) Eclipse (Eclipse Foundation, 2015) with the build automation tool Maven (Apache Software Foundation, 2015). Eclipse is open source, provides tools for refactoring, and comes with an integrated debugging environment. For code profiling, I used Java VisualVM (Oracle, 2015b), which integrates tools from the Java Development Kit (JDK).

Eclipse also provides an interface for JUnit tests, which we employed to test important basic computations, such as algorithms for geometrical computations. Though not test-driven, our development process relies on the correctness of basic parts of the software, which we tried to cover with unit tests. However, 100% of the code can rarely be covered by tests – and if possible, the effort is likely to be disproportionate to the gains. We apply unit tests where they appear necessary or helpful, especially for self-contained functionality that is used by other parts. Critical parts that are used by simulation models are reviewed by another developer after their implementation. This concerns almost all computations offered by the utility package, the core simulation loop, and the post-processing routines. The post-processing routines are highly critical because if they are flawed, the simulation model may be correct, but the analysis and hence the conclusions drawn from the results are erroneous.

2.3 Software design

In this section, I discuss the software design of the framework. We decided to develop our framework with Java and only use open source packages. The principle programming paradigm is object-oriented. The software should be as modular as possible but at the same time not impose any structure on the mathematical simulation models. We tried to compartmentalise functionality and offer it but do not impose any software pattern for the simulation models themselves. Nevertheless, a certain frame around the simulation controllers is predefined and must be, if necessary, extended to meet new requirements.

The requirements outlined in section 2.1 already indicate a certain workflow that comprises software components and artefacts (shown in figure 2.1). At first, a scenario has to be specified, which can be done with a dedicated user interface (the scenario creator) or directly in the scenario specification file. The model parameters have to be set, which is done directly in the model parameters file. These files are read by the simulator. Then the simulation is run and finally produces the simulation output (the third artefact). The output is used by the post-visualisation and the post-processing component. Both produce artefacts themselves. The post-visualisation provides functionality to take snapshots and capture videos. The post-processing component generates statistical data such as density measurements over time in a previously specified area. Technically, this requires another artefact with specifications for the post-processing, which is not shown in the figure.

For the principle structure of the software, we followed the model-view-controller (MVC) pattern (e.g., Gamma et al., 1994; Balzert, 2011). This means that the simulation state and simulated objects – the *model*, in software terms – are independent of the controlling simulation model – the *controller*. The controller, however, does depend on the model and accesses and modifies its state. The *view* both triggers the simulation as well as reads the simulation state. It depends on the software model and controller. The (software) model and controller do not depend on the view. A simplified version of the model-view-controller pattern is shown in figure 2.2. The software model is referred to as `State`, the controller `Simulator`, and the view `User Interface`.

This pattern allows for the independent development of user interfaces without changing the `Simulator` or `State` of the framework. Furthermore, the simulation state can be used by different mathematical simulation models. Sometimes, it may be necessary to extend the state with additional features for new simulation approaches, but these changes must not affect previous implementations. Therefore, this pattern is suitable for the coarse structure of the framework and meets the non-functional

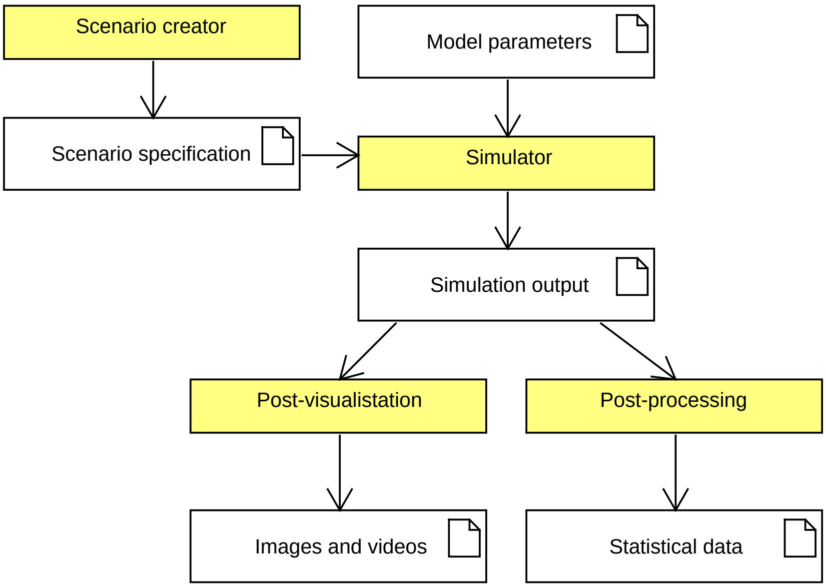

Figure 2.1: Workflow from scenario creation to output analysis through software parts and artefacts. The user has to provide a scenario specification and the model parameters to run the simulation. The simulation output can be used in the post-visualisation or for post-processing and generation of statistical data such as the density and speed of agents in a measurement area.

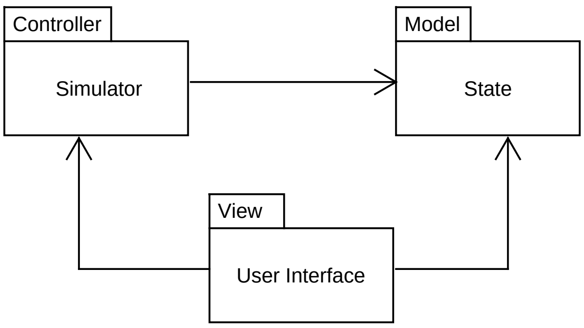

Figure 2.2: Simplified illustration of the model-view-controller (MVC) pattern for the simulation framework. The software parts model, view, and controller translate in our terminology to `State`, `User Interface`, and `Simulator`, respectively. The mathematical simulation model is located in the software part controller and not in the software part model. Different views are possible, including a command line user interface and a graphical user interface. The view has access to both the controller and model. The controller, that is, the simulator reads and modifies the state of the simulation, whereas the state is passive and does not have access to the other software parts.

requirements. Namely, it is modular, separates parts of the framework from the pedestrian model, does not impose any structure on the pedestrian model, and allows for the common use of the simulation state and views in all simulation models.

Another requirements was, that algorithms and data structures be available for all software parts and not depend on any specific model. In object-oriented programming, this extends to class structures and packages such as representations of geometrical objects. Therefore, we decided to create another package with utilities that can be used everywhere but does not depend on any other part of the software.

A generic simulation loop advances the time with a fixed time step Δt. This implements a time-slicing approach to simulation (e.g., Robinson, 2004) and seems to contradict event-driven simulation (section 4.3, chapter 4). Within the discrete time steps, events can still be arranged according to their order of occurrence, which emulates event-driven simulation. Numerical solutions of continuous models can use this generic loop by choosing small time steps Δt. Using a generic simulation loop has the advantage that the view and data processing can be called consistently after each time step (using a call-back pattern). However, this pattern *does* impose a structure on the pedestrian model by predefining the update scheme. Therefore, it may be necessary to change the implementation of the simulation loop if truly event-driven simulation – not only the same outcome but also the algorithmic implementation – is necessary. So far, this was not necessary for our applications.

Given the generic simulation loop, the models describing pedestrian movement can be implemented independently. The simulation loop calls the model to update the agent positions in the simulation. The model itself does not have access to the simulation loop. In addition to the pedestrian model, the loop also calls other controllers, including source and target controllers that add and remove agents in the scenario. The simulation loop triggers post-processing units that create simulation output. The post-processing routines do not change the state of the simulation but only read it. Therefore, they could be interpreted as a view to the simulation state. The relationships between software parts are shown in figure 2.3.

It may be argued that with an agent-based software structure arbitrary simulation models can be implemented (in section 3.4, chapter 3, I discuss agent-based models). However, this would also impose a certain software structure for simulation models. The imposed structure may impede the implementation of models that do not naturally fit into the concept of agent-based modelling, such as models based on ordinary differential equations. Models based on ordinary differential equations rather consider agents as passive elements that are moved through the environment, and the motion is described by mathematical formulas (in sections 3.2 and 3.3, chapter 3, I discuss models based on ordinary differential equations). Differential equation models are better described by vectors that contain the states of agents. While this may still be fit into an agent-based software structure, it would definitely not be the most concise and logic approach to implementing it.

Curtis et al. (2016) separated the simulation process of agents in goal selection, plan computation, and plan adaptation. The authors also referred to Ulicny and Thalmann (2002), who had used this abstraction before. This concept is fairly broad and may accommodate many models. Nevertheless, it may still not be the best representation

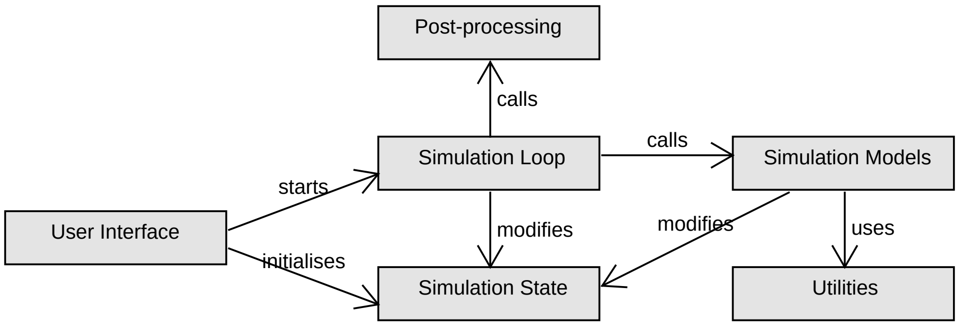

Figure 2.3: Relationship between software parts. The user interface both sets the initial simulation state and triggers the simulation loop. It may also interrupt the simulation, which is not shown here. The simulation loop calls the model to change the simulation state. It also changes the state directly by advancing the simulation time. After the simulation model has been run, post-processing is called. The simulation models use utilities, which offer general algorithms and data structures. Source and target controllers are not considered in this figure but are also triggered by the simulation loop and modify the state.

for many models that do not naturally use the same separation. For example, the optimal steps model can be implemented more concisely without it. Similar arguments hold true as for agent-based modelling.

The main purpose of our framework is to study simulation models. Implementing them in a way that represents the formal model is important for experimenting with them. A predefined structure can impede creativity in model development. In contrast, a natural implementation facilitates it and is advantageous for educational purposes. Flexible software development for models also allows for efficient algorithmic implementation. For example, parallel processing requires specific algorithms and data structures. Our framework does not predefine any agent-based software structure or, even less, impose it on models, but agent-based patterns can still be defined, offered, and used within it.

In addition to the software design, we developed a concept for the graphical user interface. The whole procedure of specifying a scenario, running the simulation, and analysing the output should be integrated in one graphical surface. Based on legacy code for individual solutions, an interface for the management of various scenarios and outputs was available. A graphical interface for the creation of topographies and a post-visualisation should be integrated in the graphical user interface. The result is presented in the next section.

2.4 Functionality

In this section, I briefly review the functionality of the framework and demonstrate that the functional requirements defined in section 2.1 are met. The principle function of the software is to simulate microscopic pedestrian dynamics. Agents are represented as

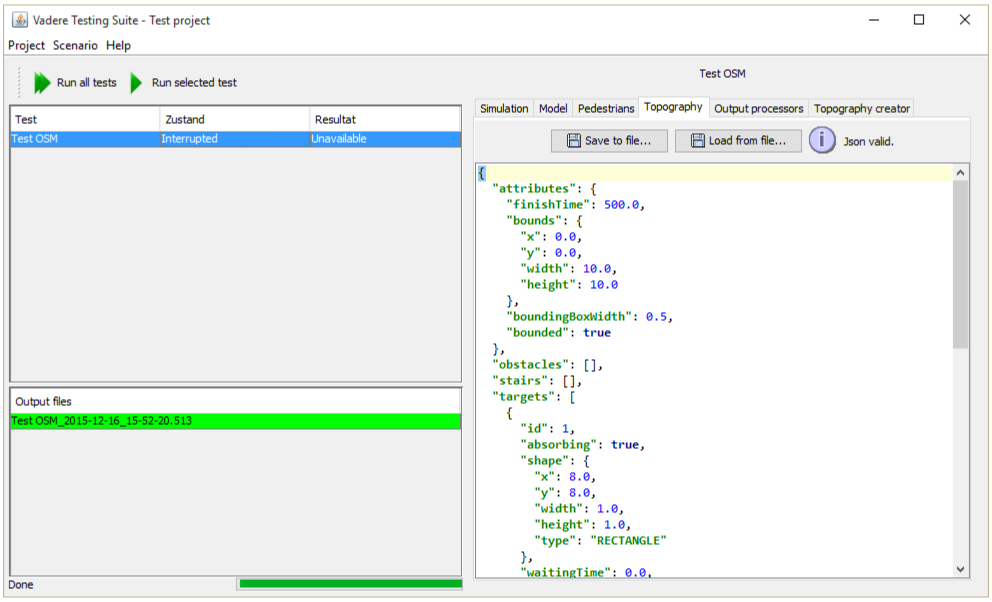

Figure 2.4: Vadere graphical user interface. The top left side of the interface provides a list of scenario definitions. Below that, a list of output files is displayed. Output files can be opened and contain all parameters. The GUI has an integrated post-visualisation and graphical scenario creator.

two-dimensional circles (other shapes could easily be implemented). They are created in a source or at arbitrary predefined positions in the scenario. They either have no target and only move based on the influences of other agents and obstacles or try to reach a target. Their behaviour is described by simulation models that are the focus of later chapters in this work. The scenarios and model parameters have to be set before the simulation begins. During and after the simulation, the simulation outcome can be recorded and processed for later analysis.

A typical use case begins with the specification of a scenario and the model parameters. Both can be considered parameters to the simulation. We separated the specification of the model, the topography, the simulated pedestrians (agents), and other simulation parameters. Together, they form the input to the simulation and determine which model is used. Given the software and the parameters, the simulation can be exactly replicated assuming a random seed was set.

The whole specification is stored in one file using the open standard format JSON ("JavaScript Object Notation", JSON Contributors, 2015). JSON is a widely used standard with minimal syntactic overhead. It can easily be learned and read, also by users who are not proficient in programming. We decided against XML as it requires more notation, and parsing XML in Java is less convenient than parsing JSON. For each part of the specification, a separate JSON block is defined.

Code listing 2.1 provides an example of a small topography definition in JSON. Names are on the left in quotation marks and values on the right, separated by a colon. Values can be numbers (in blue), text (in red), lists of the same type of values (in square brackets), or other name-value pairs (in curly brackets). For example, *obstacles* is a list of obstacle definitions, and each obstacle has a geometrical shape. Sources and targets have a similar structure with additional parameters. For example, the source stores a list with IDs of the targets the agent has to reach in the same order.

Listing 2.1: Example of a topography definition in JSON. Information is structured in a list of name-value pairs. The names are on the left, and the values on the right (separated by a colon). Values can be lists of other values, and name-value pairs can also be nested. In this case, numerical values are given in metres and seconds.

```json
{
  "attributes": {
    "finishTime": 500.0,
    "bounds": {
      "x": 0.0,
      "y": 0.0,
      "width": 10.0,
      "height": 10.0
    },
    "boundingBoxWidth": 0.5,
    "bounded": true
  },
  "obstacles": [
    {
      "shape": {
        "x": 5.0,
        "y": 2.0,
        "width": 2.0,
        "height": 2.0,
        "type": "RECTANGLE"
      },
      "id": 1
    },
    {
      "shape": {
        "x": 1.0,
        "y": 7.0,
        "width": 2.0,
        "height": 2.0,
        "type": "RECTANGLE"
      },
      "id": 2
    }
  ],
  "targets": [
    {
      "id": 1,
      "absorbing": false,
      "shape": {
        "x": 1.0,
        "y": 1.0,
        "width": 3.0,
        "height": 3.0,
        "type": "RECTANGLE"
      },
      "waitingTime": 0.0,
      "waitingTimeYellowPhase": 0.0,
```

```
      "parallelWaiters": 0,
      "individualWaiting": true,
      "deletionDistance": 0.1,
      "startingWithRedLight": false
    }
  ],
  "sources": [
    {
      "id": 1,
      "shape": {
        "x": 5.0,
        "y": 5.0,
        "width": 4.0,
        "height": 4.0,
        "type": "RECTANGLE"
      },
      "spawnDelay": 1.0,
      "spawnNumber": 10,
      "startTime": 0.0,
      "endTime": 0.0,
      "spawnAtRandomPositions": false,
      "useFreeSpaceOnly": false,
      "targetIds": [
        1
      ],
      "dynamicElementType": "PEDESTRIAN"
    }
  ],
  "pedestrians": []
}
```

The whole specification of a simulation run can be carried out in a text editor, which was one of the functional requirements. Additionally, we developed a graphical user interface (GUI) that integrates a JSON editor and separates the various parts of the scenario and model definition. A snapshot of the graphical interface is shown in figure 2.4. The GUI offers a topography creator and a post-visualisation tool. The topography creator allows for visually creating and editing the scenario. The scenario specification is encoded into the JSON format and displayed directly. Multiple scenario files can be managed (on the left), and the respective output files can be opened in the post-visualisation.

Online visualisation allows for the immediate visual validation of a simulation run. The simulation can be aborted if there is an obvious mistake causing erroneous output – a feature which is very time saving. The online visualisation provides information on the current progress of the simulation and is imperative for validation. Visualisation does not provide a proof of correctness in terms of validity, but it is crucial to detect errors in the implementation or aid in basic validation (Gipps, 1986). Without it, some problems are very hard to detect in numerical data and might never be spotted.

The GUI provides a wizard for the specification of output processors. The output processor definition is also stored in the JSON format and can be specified in a text editor. The format of the data generated by the output processor is part of the specification and may be adjusted according to the data analysis requirements.

The simulation framework meets the functional requirements defined in section 2.1. Namely, the framework runs simulations of pedestrian crowd behaviour. Parameters are specified and stored in text files with the JSON format. The GUI allows for the graphical creation of topography layouts and provides an integrated text editor for the specification of all parameters. The framework stores output that describes the trajectories of individual agents. Statistical data is generated in online processing modules. With the integrated post-visualisation, the simulation output can be analysed visually.

2.5 Software architecture

I give a more detailed description of the software architecture and how it meets the non-functional requirements in this section. The architecture was developed based on the design presented above. In the implementation, some details are not as emphasised as in the design but are still compatible with it. At first, I describe the overall package structure that has already been introduced in the software design (section 2.3). Secondly, the topography class and its dependencies are outlined. Thirdly, I describe the controllers that change the topography and the simulation models, which can also be seen as controllers. Relevant software design patterns are considered. Finally, I demonstrate how the main simulation loop uses all these parts and how it is related to them in the class structure.

According to the non-functional software requirements (section 2.1), only open source software was used for the framework. The framework is written in Java, which is a platform independent, high-level, object-oriented programming language. Throughout the development, special attention was paid to modularity, reusability, and the independence of simulation models.

Figure 2.5 shows the principal package structure that separates the framework into parts similar to the model-view-controller pattern (section 2.3 and figure 2.2). In the following, the software model is referred to as `State`, the software controller `Simulator`, and the view `GUI`. The GUI has access to both the classes of the simulator and the state. Through the graphical interface, the user can start and stop the simulation. The GUI offers an interface to modify the simulation parameters that are stored in the state. The simulator reads these parameters from the state, builds the objects necessary for the simulation, and modifies their state in the course of the simulation.

The elements contained in the package `Simulator` are the controllers, which modify the simulation state, more specifically, the topography. The simulation models are a special type of controller that include the mathematical description of the behaviour of agents. The scenario element controllers create and remove agents from the topography. Post-processing controllers and routines handle the input and output. The state holds the parameters, which were set in the GUI and are referred to as `Attributes` in the software implementation. `ScenarioElements` describe the topography of the simulation. The GUI has a parameter editor, scenario creator, an online and post-visualisation. I do not discuss the GUI in detail as it does not belong to the core of the simulation framework. Nevertheless, it is important for the usability of the

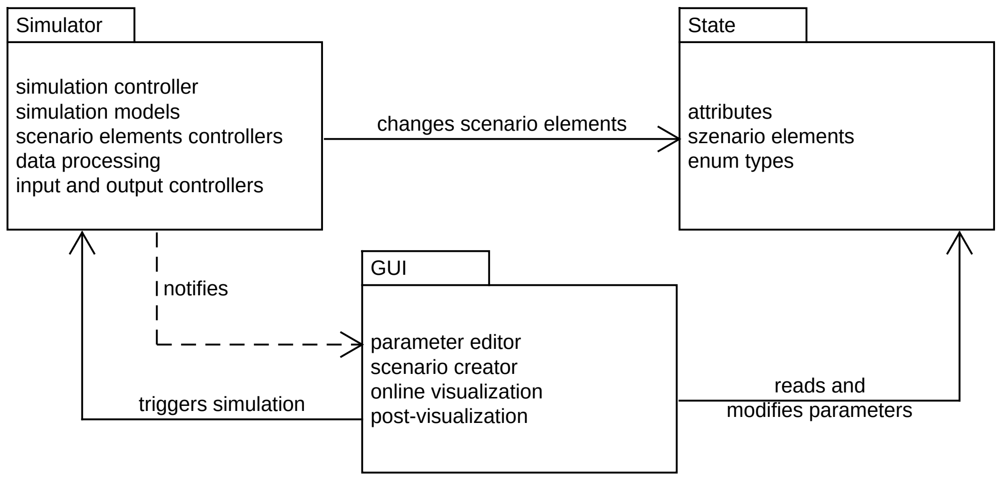

Figure 2.5: Vadere design pattern for the simulation framework. The architecture is inspired by the model-view-controller (MVC) pattern (section 2.3). The `State` holds both the attributes, which can be modified in the `GUI`, and the scenario elements, which are created and used by the `Simulator`. The simulator controls the state. The GUI is notified by the simulator when the simulation state has changed, and then reads and displays the simulation state in the online visualisation.

software, and a visualisation is an essential tool for verification and validation.

The separation of the three packages ensures modularity on a coarse level. This is important to meet the requirements listed in section 2.1, namely, the modularity and reusability of basic algorithms and data structures. The packages `State` and `GUI` usually do not have to be changed much when new models are introduced and normally do not have to be changed at all when existing models are altered. The separation of the three packages does not impose any structure on the simulation models themselves.

Figure 2.6 shows the structure of the state in more detail. Every element, simulation model, and processor has an additional class `Attributes`. The attributes can be set in the GUI or directly in a text editor with the JSON notation described in section 2.4. They are later used to create the simulation model and represent the data structure of the parameters. The scenario elements are defined in the state too with dedicated classes, which are interface definitions for scenario elements (all elements), dynamic elements (moving elements), and concrete implementations. Concrete class implementations include `Pedestrian`, `Target`, `Source`, and `Obstacle`. Obstacles do not actively change the scenario or contain any routines of the simulation model. Finally, several types are defined and collected in the state.

In addition to the packages described in the model-view-controller pattern, we maintain one package for other utilities. The package `Utils` can be used by all other packages but does not depend on any of them itself. It provides classes for geometrical objects that are implementations of the interface `Shape`, such as `Rectangle` and `Polygon`. Every source or target has one shape object that describes its geometrical representation and position in the scenario. Furthermore, the package `Utils` collects

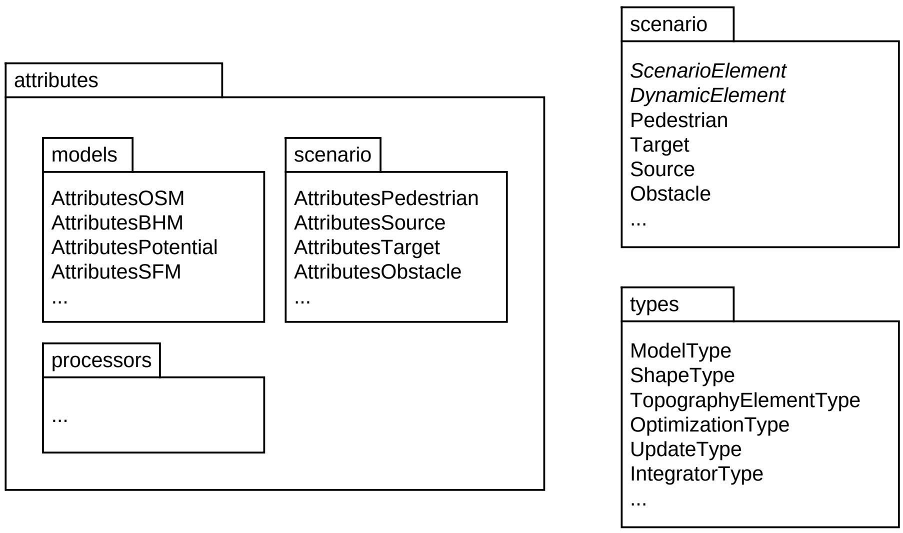

Figure 2.6: State package diagram. Attributes represent the data structure that stores input parameters for later use in the simulation. For example, the classes `AttributesOSM`, `AttributesBHM`, and `AttributesSFM` store the parameters for the respective pedestrian motion models. The package `Scenario` contains a series of classes that is used by the class `Topography`. The package `Types` contains enum classes that define pedestrian models, geometrical shapes, and scenario elements. For example, `ModelType` has values such as `GRADIENT_NAVIGATION_MODEL` and `OPTIMAL_STEPS_MODEL` for each pedestrian motion model.

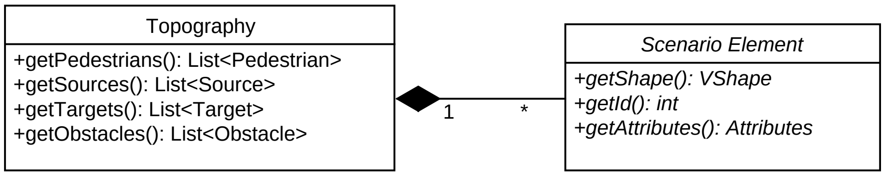

Figure 2.7: Topography class diagram. The object of type `Topography` contains all elements in the scenario. It provides methods that return the respective elements. The topography does not change its state but provides methods for that purpose which can be called from controllers.

data structures and numerical computations, such as cell lists. Cell lists is a spatial data structure for determining neighbours in the proximity (Allen, 1987, section 5.3). In contrast to iterating through all other objects, the complexity of cell lists does not depend on the total number of other objects in the scenario. With the utilities package, the requirement of reusable code, especially numerical computations, is met.

An object of type `Topography` contains all elements present in the scenario (figure 2.7). The various controllers and the simulation model modify the topography object. The topography never changes itself, and, therefore, it is passive, which is in accordance with the notion of a software model. The interface `ScenarioElement` only requires basic access methods for the shape object it contains, its attributes, and its ID.

`ScenarioElement` implementations are `Source`, `Target`, and `Obstacle` (figure 2.8). Each of these classes provides the information for a specific functionality in the simulation. Sources create agents, targets attract agents and remove them from the scenario, and obstacles are inaccessible areas that do not actively affect the agents, but the agents do consider them in their behaviour. The classes in the package `State` only provide the information on what these objects are meant to do and do not change the topography. Instead, the topography is changed by the respective controllers in the `Simulator`, which use the information. Therefore, the information stored in the scenario elements indirectly affects the topography.

`DynamicElement` is a derivative of `ScenarioElement` and represents objects that actively move in the simulated environment (figure 2.8). These are agents of the type `Pedestrian`, but other implementations such as `Car` are possible and fit into the class hierarchy. `Pedestrian` is a concrete class that can be used directly, but it can also be derived in a simulation model implementation if an agent-based perspective is preferred. For example, the optimal steps model and the heuristic approach to pedestrian behaviour both implement their own derivative of `Pedestrian`. In these cases, the pedestrian objects are not passive but also contain routines that describe their behaviour. This derived, active pedestrian is not part of the state any more but of the simulation model in the simulator.

The classes belonging to the topography are used by the GUI and in the simulation models. Usually, this collection of classes does not have to be changed when new models are introduced. Therefore, this part of the framework is largely independent

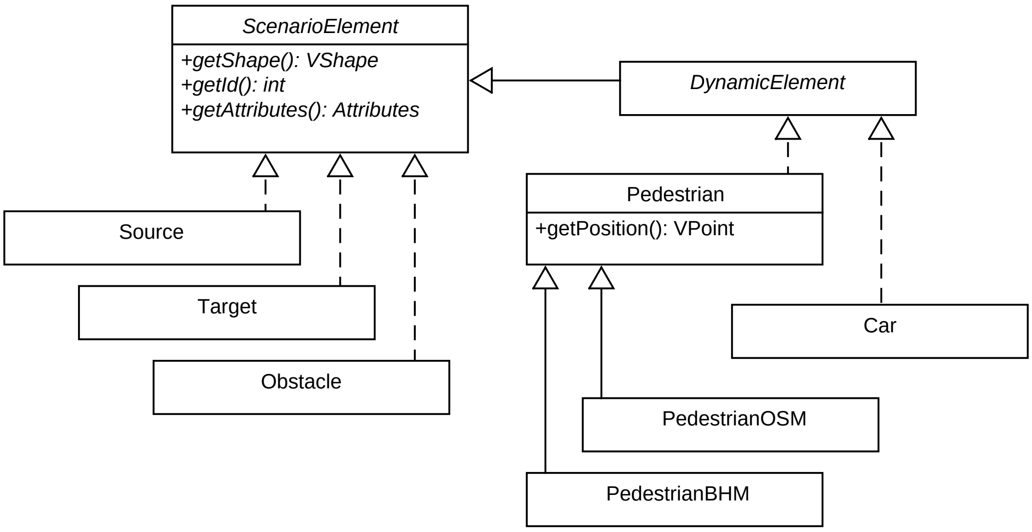

Figure 2.8: Class diagram of `ScenarioElement` and its derived classes. The objects of type `Source`, `Target`, and `Obstacle` embed specific information on what these elements do in the scenario. This information is used by the respective controllers. Objects of type `DynamicElement` are elements that actively move in the simulated environment. `Pedestrian` is the most important implementation, but other implementations such as `Car` are possible. For some models, it can be convenient to have an agent representation that contains the description of their behaviour. For this, the class `Pedestrian` can be derived and used in the `Simulator`. For example, the optimal steps model (OSM) has its own derivative called `PedestrianOSM`.

and reusable, which were important requirements in section 2.1. The classes also do not impose any structure on the modelling approaches. In some cases, additional scenario elements may be necessary for new simulation scenarios or models, for example, a model of stairs. Due to the generic interface `ScenarioElement`, extending the existing collection of concrete scenario elements is facilitated.

The classes `Source` and `Target` have controllers that read the information stored in them and carry out the encoded tasks, that is, create and remove agents in the scenario. The scenario elements and the topography in the state do not have access to their respective controllers but provide methods for them. The source controller is generic and does not know what type of agent it has to create. The simulation model, which contains the description of the agent behaviour, has to provide the specification of the agent. For this, we use a factory pattern (e.g., Gamma et al., 1994) that creates concrete objects of type `DynamicElement`. The interface `OperationModel` is derived from the factory specification and must be implemented by concrete simulation models, such as the social force model and the optimal steps model. The source controller calls the factory whenever it has to create agents in the scenario, receives the objects, and adds them to the topography. With this pattern, the topography and the source controller are independent of the simulation model. New models can be added without changing or adding code to the state or generic simulation routines. The whole pattern is shown in figure 2.9.

In addition to being a factory for the generation of agents, the operation model is also an active callback. The interface `ActiveCallback` requires methods that can be called by the simulation loop. These methods represent the interface between the generic simulation loop and functionality and the simulation model, which determines the behaviour of agents. Therefore, every simulation model has to implement the interface `OperationModel`. The class structure is shown in figure 2.10. This architecture allows for the introduction of new models without changing the general structure of the framework, which was one of the software requirements.

The whole parameter specification for the simulation model has to be given in the JSON format. The JSON source is interpreted, and the required objects are created in the framework. For this process, it is necessary that the software knows how to create the simulation model. The class `ModelCreator` (figure 2.11) encapsulates the necessary logic and routines for the process of building the simulation model. This is also necessary because some models may use other models and hence must be created in a predefined order. The order is specified by the model creator and later used by the simulation loop. The model creator is not generic and has to be adapted whenever a new model is included in the framework.

The object of type `Simulation` uses all components defined so far. It is generic and does not have to be changed when new models are implemented. Figure 2.12 shows its class diagram with the relations to the other classes. The simulation object holds one topography object, one topography controller, and an arbitrary number of source and target controllers. In every simulation loop, the simulation iterates over all active and passive callbacks it stores. The active callbacks can change the state of the simulation, that is, the topography object. The order of the active callbacks is predefined by the model creator and encoded in the order of the list they are held

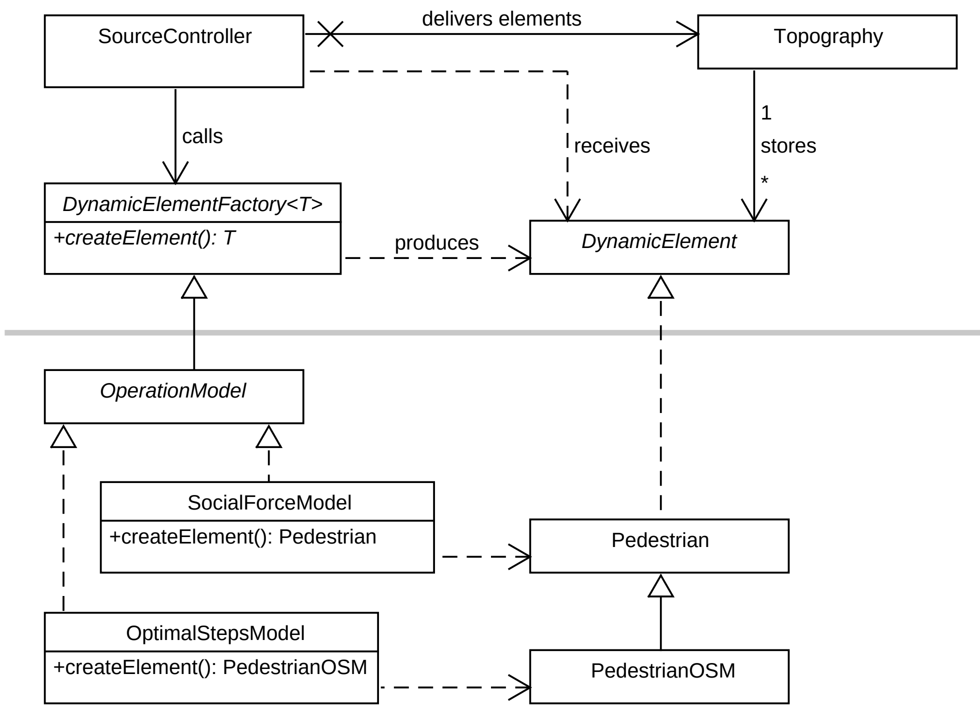

Figure 2.9: Class diagram of `DynamicElementFactory`. The class `SourceController` calls an implementation of `DynamicElementFactory`, which generates objects of the type `DynamicElement`. It then delivers the objects to the topography. Operation models, such as the optimal steps model, implement the factory and create objects of type `Pedestrian` or objects derived from this class. With this pattern, new simulation models can be added without changing the code of the generic simulation routine, especially the scenario controllers and the state.

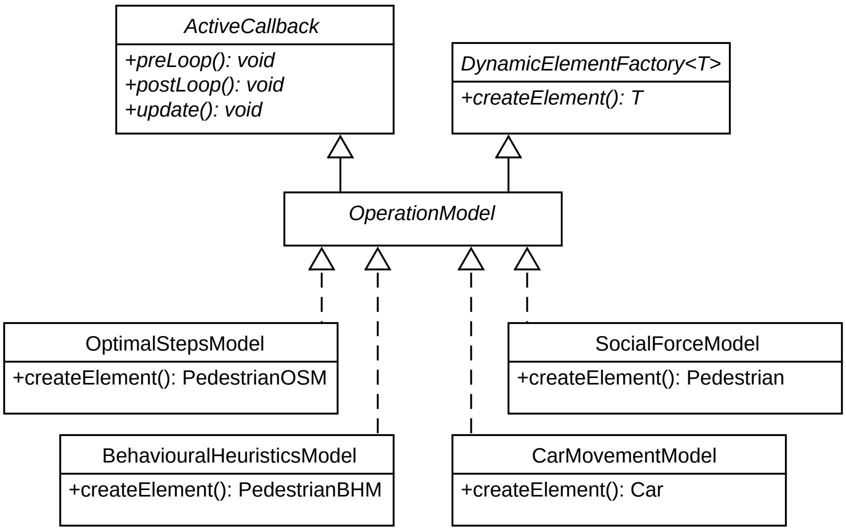

Figure 2.10: `OperationModel` class diagram. The interface `OperationModel` inherits the requirements from both the interface `ActiveCallback` and `DynamicElementFactory`. It represents the interface between the generic simulation loop and routines and the concrete simulation model implementations, such as the class `OptimalStepsModel`. Therefore, every simulation model has to implement the requirements given by the interface `OperationModel`.

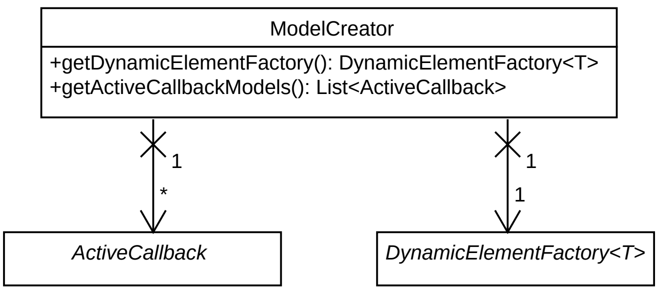

Figure 2.11: `ModelCreator` class diagram. The model creator builds the simulation model objects, which are active callbacks and one object of type `DynamicElementFactory`. The model creator ensures that all required parts are created and embeds the order of creation.

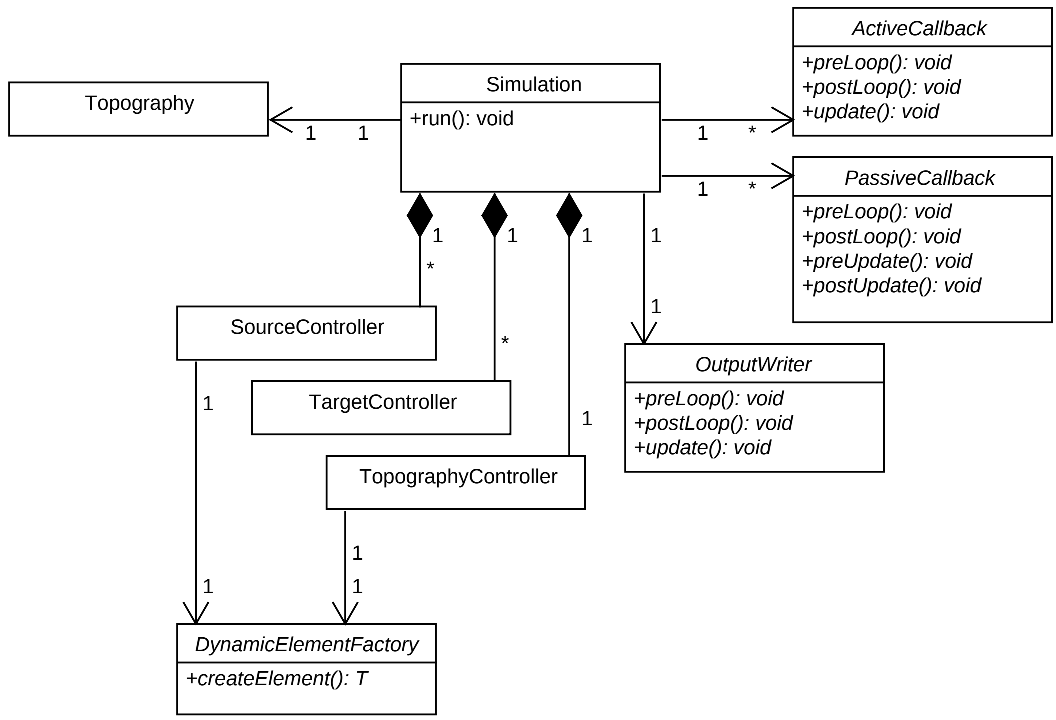

Figure 2.12: `Simulation` class diagram. The simulation calls the active callbacks in a predefined order. Passive callbacks and output writers do not change the state of the simulation and thus do not have to be ordered explicitly. The objects of type `SourceController`, `TargetController`, and `TopographyController` are called in the simulation loop before the active callbacks. The class `Simulation` is generic and depends on the chosen simulation model only through the interfaces `ActiveCallback` and `DynamicElementFactory`.

in. The most important active callbacks are the simulation models. Passive callbacks and output writers do not change the state of the simulation and thus do not have to be ordered explicitly. The output processors are implemented as passive callbacks.

The procedure of the simulation main loop is illustrated with an activity diagram in figure 2.13. The simulation needs the topography, attributes, scenario elements, controllers, active and passive callbacks, and the dynamic element factory. The callbacks have to implement a method named `preLoop`, which is called before the start of the simulation, and a method `postLoop`, which is called after the simulation run. Passive callbacks have additional methods that are called before and after each loop iteration. At first, the source and target controllers are called. Then, the active callbacks can change the scenario, their order being specified in the list they are held in. After that, output writers are triggered before the simulation time is incremented. If the simulation has come to an end according to some criterion, the `postLoop` methods are called. Otherwise, the loop is entered again.

The simulation loop builds on the previously defined classes. They were all design with the specific goal of modularity and independence of the concrete simulation models. When introducing new models, the simulation loop, topography with scenario elements, GUI, and the scenario element controllers do not have to be changed. That means, the respective parts of the software are modular and reusable, and when implementing new simulation models, the framework does not have to be changed. Finally, the framework does not impose much structure on the simulation models and can easily be extended if additional scenario elements are required. Therefore, the architecture meets the requirements and the design from sections 2.1 and 2.3.

2.6 Utilisation and future directions

The software framework has already been used for teaching seminars on computer simulation and mathematical modelling. Students writing their Bachelor's and Master's theses have used the framework to conduct simulation studies, to extend and investigate existing simulation models, or to develop their own model. For example, the implementation of a model of granular flow has demonstrated the flexibility of the framework. One student investigated a scenario where pedestrians interact with car traffic at a crossroad scenario. For this, he implemented a velocity-based car simulation model in Vadere.

We intend to publish the framework as an open source project. This would allow other research groups to directly use our software for their own studies. Simulating pedestrian dynamics always relies on a software implementation, which can be time-consuming and challenging to realise. The availability of the software with the source code eases collaboration for studies and publications and allows for the reproducibility of results. Therefore, we consider an open-source publication of the software framework Vadere a worthwhile scientific contribution.

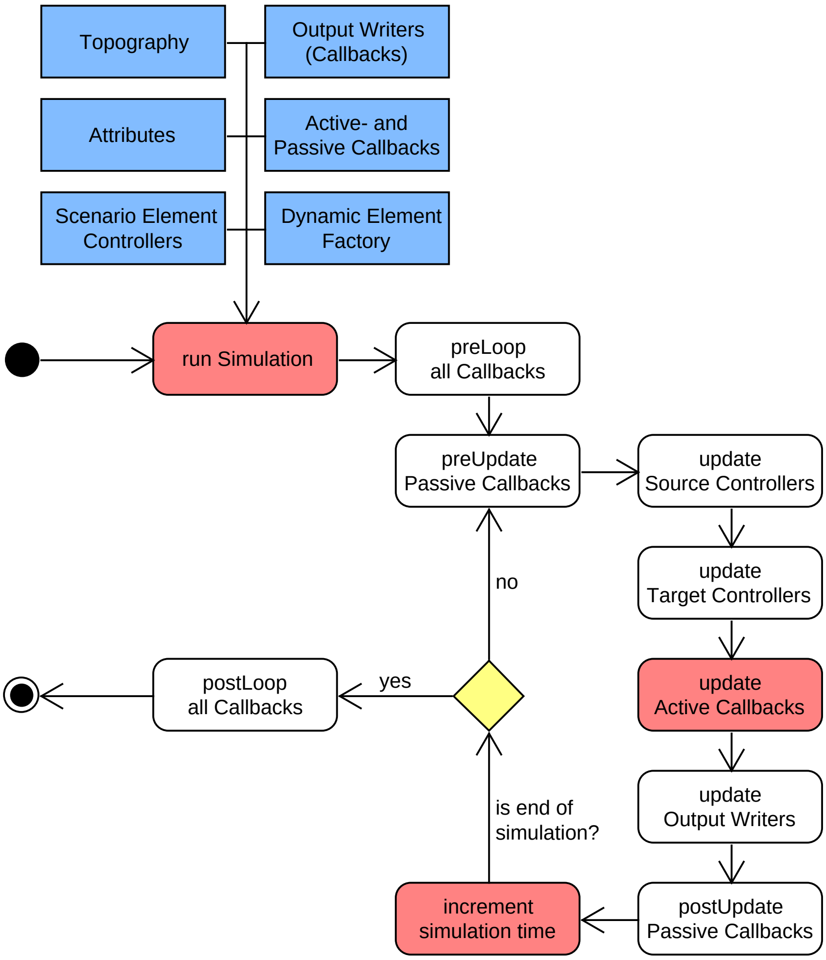

Figure 2.13: Simulation activity diagram. Objects that are provided for the simulation are shown in blue rectangles. The decision whether the simulation is continued or stopped is represented with a yellow diamond. The start of the programme is shown as a black filled circle and the end as two concentric circles with the inner one being filled. The simulation loop calls scenario controllers and active and passive callbacks in a predefined order.

2.7 Summary

In this chapter, I presented and discussed the software framework Vadere, which we developed for the purpose of studying pedestrian crowd simulation models. I identified principal requirements that address the objective of scientific model development and discussed other open source software projects for the simulation of pedestrian dynamics, which did not meet our demands.

I specified the requirements for our framework in section 2.1. The principle functional requirement was to run microscopic simulations of pedestrian crowd behaviour. The non-functional requirements were just as important because the framework specifically targets the development and study of new models. The software design had to be flexible so that it allows for fast changes and new developments.

Based on the requirements, I argued for an agile development process in section 2.2. Because of the necessary flexibility, the software architecture could only be predesigned to a certain degree. We used a toolchain that supports this process and allows for collaborative software development. Namely, we used the distributed version control system Git and the IDE Eclipse complemented with other tools like Maven and Java VisualVM.

The software design aimed at modularity and flexibility (section 2.3). The principle structure of the framework was inspired by the model-view-controller pattern and separates the generic simulation routines from the concrete simulation models that determine how agents in the simulation behave. Additionally, a package with utilities for generic computational tasks was provided. The design must allow for unobstructed model development and, therefore, cannot impose any software structure on them.

After the specification of the requirements and design, I described how we realised this concept in terms of software functionality (section 2.4) and architecture (section 2.5). Vadere provides functionality for the specification of simulation parameters, management of simulation runs, and visualisation and analysis of the simulation output. The parameters that determine the simulation are specified in JSON and can be edited directly with a text editor. A graphical user interface offers a graphical scenario creator and separate sections in which the parameters can be viewed and changed. The simulation output can be viewed in an online and post-visualisation. The principle architecture is based on the software design, which emphasises genericity of the simulation loop and the topography definitions. New models can be implemented without changing this part of the framework. The simulation loop accesses and uses the simulation model classes through predefined interfaces. I argued that the requirements and the software design are met with this functionality and architecture.

Finally, I described how the software framework has been used for research and teaching in section 2.6. The various model extensions and implementations that have been realised in Vadere demonstrate that the architecture is flexible and suitable for research. To allow others full access to the software, we intend to publish it as open source in the future.

Chapter 3

Modelling approaches

I review modelling approaches that are particularly relevant in microscopic pedestrian crowd simulation in this chapter. The focus lies on the conceptual features, but emergent phenomena are also considered. I assess and classify the models by their use of scalar fields (Seitz et al., 2016). Scalar fields are a concept used in many of the approaches, although they are usually not called that way explicitly. The perspective of scalar fields, also referred to as the *superposition principle*, is a starting point from which similarities and differences can be explored.

The aim of this chapter is to discuss known approaches, critically review them, and assess their suitability for representing the underlying processes of pedestrian dynamics. This discussion is particularly relevant for the next chapter, which describes the optimal steps model. The optimal steps model builds on the concept of scalar fields but otherwise departs from previous approaches. In part II of this work, I argue that none of the existing approaches meets the requirements of a plausible process-oriented simulation (Moussaïd and Nelson, 2014). However, every approach has its own interesting modelling mechanisms and may meet the demands of some application. Considering the historical context, the authors of the respective studies have laid the foundation for new developments.

All approaches discussed in this chapter simulate pedestrian dynamics in the two-dimensional transverse plane (top-down view). With some exceptions, simulated pedestrians (agents) are represented with a circular shape. Obstacles are abstract shapes that cannot be stepped on, sources are areas where agents are created, and targets are areas agents try to reach. The origin-destination (OD) relations from the sources to the targets are known, which means that the distribution of agents created in the sources and the walking destinations are given. Both the source and the target are simple geometrical shapes. In some cases, every agent may be placed at a specific position at the beginning of the simulation instead of being created in a source over time.

Planning and decision making on a larger spatial scale are out of scope in this chapter, which is, in principle, also the case for the rest of this work. Two exceptions are navigation fields and graph-based routeing, which allow for navigation around obstacles in large scenarios. I treat both of these approaches mainly as background information and hence do not discuss them in detail. I also do not intend to provide a full review of published models, because there are reviews available by various authors

(Zheng et al., 2009; Papadimitriou et al., 2009; Duives et al., 2013). In addition, Bellomo and Dogbe (2011) review the literature from a mathematical perspective and Templeton et al. (2015) from the perspective of social psychology.

When choosing a model, many choices have to be made. To facilitate these choices and to get an overview of the available options, some classification can be helpful. Zheng et al. (2009) identify seven classes: cellular automata, lattice gas models, social force models, fluid-dynamic models, agent-based models, game-theoretic models, and "approaches based on experiments with animals". In contrast, I classify models by *what is being modelled* rather than by analogies used for them. Three major classes can be found. The first class are cellular automata, which are determined by rules that describe the transition from one state to the next (section 3.1). Other approaches, such as the optimal steps model, have a similar discrete process. The second class are velocity-based models (section 3.2). Here, the velocity (the speed and direction) of agents is directly described by a first-order ordinary differential equation. *What is being modelled* is the velocity. The third class are force-based models (section 3.3). In these models, the forces acting on agents are manipulated to obtain motion. Mathematically, a second-order ordinary differential equation has to be solved to obtain the positions of agents.

This coarse classification already determines other aspects of the simulation model. Rule-based models are usually discrete in time and space. Velocity and force-based models are continuous, although discretisation is still necessary for numerical computation. Other choices remain to be taken, for example, whether a deterministic or probabilistic approach is chosen. Most models have the common basis of scalar fields, which I investigate in section 3.5. I also discuss some emergent behaviours of models that use scalar fields in the same section.

Looking beyond the concept of scalar fields and the three classes described, other models can be found (section 3.4). In computer animation and game development, agent-based modelling is well-known, although it is not always easy to distinguish this class from others. In agent-based models, the individual's goals are the main focus, and hence, sometimes a physical layer or physics engine realises the actual locomotion in the simulated environment. This is also the case in models that aim to describe perception and decision making in agents, which I summarise as cognitive models. Both approaches seem to allow for new modelling possibilities, such as anticipatory behaviour. Agent-based models can become difficult to validate for applications that require accurate predictions of density, speed, and other measures.

Finally, apart from the locomotion layer, path planning can be realised with many different algorithms and models of decision making. For example, navigation fields can be used to represent the shortest, quickest, or most attractive path (Kretz, 2009; Hartmann, 2010). As an alternative, a variety of graph-based approaches has been presented to solve similar tasks (Arikan et al., 2001; Kneidl et al., 2012; Kneidl, 2013). I focus on motion and interactions in the proximity and hence do not discuss strategic decision making and route choices in detail.

3.1 Cellular automata

Cellular automata can be used for many applications – traffic and pedestrian simulation has only later been one of them. "Cellular automata are examples of mathematical systems constructed from many identical components, each simple, but together capable of complex behaviour" (Wolfram, 1984). An outline of the historical origin was also given by Wolfram (1983):

> Cellular automata were originally introduced by von Neumann and Ulam (under the name of "cellular spaces") as a possible idealization of biological systems (von Neumann, 1963, 1966), with the particular purpose of modelling biological self-reproduction. They have been applied and reintroduced for a wide variety of purposes, and referred to by a variety of names, including "tessellation automata," "homogeneous structures," "cellular structures," "tessellation structures," and "iterative arrays." (Wolfram, 1983)[1]

Cellular automata were first established for car traffic by Nagel and Schreckenberg (1992) and Biham et al. (1992). This approach has since been thoroughly studied (Schadschneider and Schreckenberg, 1993; Schreckenberg et al., 1995) and extended (Emmerich and Rank, 1997; Nishinari et al., 2004), for example, with two-lane traffic (Rickert et al., 1996; Nagel et al., 1998). It has also been applied, for example, in the online simulation of car traffic (Wahle et al., 2001). There may be certain similarities among car traffic and pedestrian stream models, but in principle, car traffic is studied as a one-dimensional system, whereas pedestrian dynamics is mostly studied as a two-dimensional system. As I discuss below, two-dimensional cellular automata for pedestrian dynamics produce artefacts that are not present in the one-dimensional setting. Therefore, the model development is more challenging for pedestrian simulations than for car traffic simulations.

Given the variety of applications cellular structures can be used for, it is important to identify a common basis. The following definition also largely holds true for most cellular automata in pedestrian dynamics:

> Cellular automata are mathematical idealizations of physical systems in which space and time are discrete, and physical quantities take on a finite set of discrete values. A cellular automaton consists of a regular uniform lattice (or "array"), usually infinite in extent, with a discrete variable at each site ("cell"). The state of a cellular automaton is completely specified by the values of the variables at each site. A cellular automaton evolves in discrete time steps, with the value of the variable at one site being affected by the values of variables at sites in its "neighborhood" on the previous time step. The neighborhood of a site is typically taken to be the site itself and all immediately adjacent sites. The variables at each site are updated simultaneously ("synchronously"), based on the values of the variables in their neighborhood at the preceding time step, and according to a definite set of "local rules." (Wolfram, 1983)

[1] The references in the quote correspond to (von Neumann, 1963, 1966).

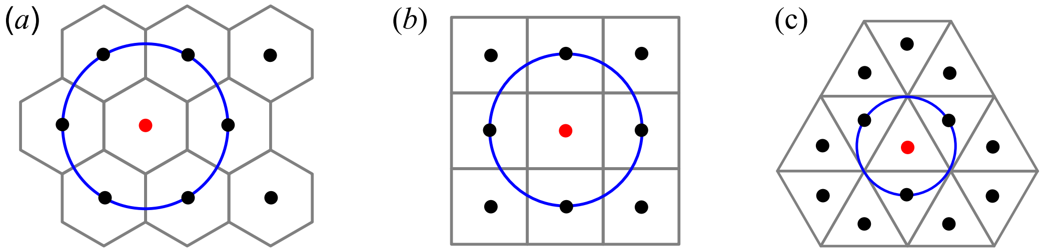

Figure 3.1: Cellular automata with different grids: hexagonal grid (*a*), rectangular grid (*b*), and triangular grid (*c*). The (black) points depict the centres of cells and the red point the focal position. For each case, the (blue) circle indicates how the movement options could be emulated in the optimal steps model given a suitable discretisation. (Figure: Seitz et al., 2016)

Boccara (2010, chapter 6) gave another definition of general cellular automata and studied them from a formal perspective.

Following the definition by Wolfram (1983), some models discussed below are not, strictly speaking, cellular automata. Specifically, some of the models do not update the sites simultaneously, and, more often, the values of sites are determined by more than just the adjacent sites. The rationale behind the motion process of pedestrian agents also somewhat contradicts the idea of cellular automata in general. In pedestrian dynamics, not the state of the cell is of interest but the position and motion of the agent. I use the term *cellular automaton* loosely for all models in which agents are represented in a grid of cells and point out where this deviates from the above definition.

At first, I give a short description of how cellular automata can be used for the simulation of pedestrian flows. In figure 3.1, the possible grid structures for two-dimensional cellular automata are shown: hexagonal cells (Maniccam, 2003; Hartmann, 2010; Leng et al., 2014), rectangular cells (Gipps and Marksjö, 1985; Blue et al., 1997; Burstedde et al., 2001), and triangular cells (in figure 3.1 *a*, *b*, and *c*, respectively).

Rectangular cells are most common for applications in pedestrian dynamics. Two options for defining the neighbourhood and thus the adjacent cells exist with rectangular cells: the von Neumann neighbourhood includes only the four cells with a common side; the Moore neighbourhood additionally includes diagonal cells connected with a common vertex. Hexagonal cells have the advantage that all adjacent cells have the same distance. Triangular cells only have three adjacent cells with a common side, although nine additional neighbours could be defined over the vertices. To my knowledge, no cellular automata with triangular cells has been presented for pedestrian dynamics so far, which may indicate that there are no advantages over the other two options.

Another major distinction among cellular automata for pedestrian dynamics is whether they are based on a deterministic (Gipps and Marksjö, 1985; Blue et al., 1997; Fukui and Ishibashi, 1999) or a probabilistic (Muramatsu and Nagatani, 2000; Burstedde et al., 2001; Tajima and Nagatani, 2001; Klüpfel, 2003) scheme. Furthermore, there are great differences among models within these two categories. I describe several approaches in the following, mostly in the order of their publication.

Gipps and Marksjö (1985) published the first model that used a cellular grid for pedestrian dynamics. To my knowledge, it was also the first simulation model presented for pedestrian dynamics in general. The authors separated the route choice from movement along a selected route, which is still a common practice and allows for a dedicated decision process for both aspects. In the following, I focus on the movement scheme without the route choice mechanisms. The model can be classified as a cellular automaton because it uses a grid of squares, only one agent can occupy a cell, and agents can only remain at the current position or move to an adjacent cell. Nevertheless, some argue it is "not quite" a cellular automaton (Blue and Adler, 2001). The most important aspects are described by the authors in the following paragraph of their paper:

> The study region is divided into squares 0.5 metres on each side by a rectangular grid. Each cell can be occupied by at most one pedestrian, and a score is assigned to each cell on the basis of its proximity to pedestrians. This score represents the repulsive effects of nearby pedestrians, and has to be balanced against the gains made by the subject in moving towards his destination. Where the fields of two pedestrians overlap, the score in each cell is the sum of the scores generated by the pedestrians individually. (Gipps and Marksjö, 1985)

The score for cells can be interpreted as a scalar field (section 3.5). Hence, this publication laid the basis for later approaches, including the social force model, that use the idea of attractive and repulsive contributions to a value describing the attractiveness of positions around the agent. While the authors referred to the repulsive effects of other agents as *forces*, the net benefit they calculate for each cell based on the score and the gain from getting closer to the target can well be interpreted as utility. This interpretation seems appropriate as agents choose the adjacent cell with the best score, which can be interpreted as utility optimisation.

In the same paper, the authors identified the problem that agents will prefer diagonal movement steps because of the cell structure. They proposed to remedy this problem "by making the gain from the move dependent on the angle of deviation from the desired path rather than the distance of the target cell from the destination" (Gipps and Marksjö, 1985). They also discussed the problem of agents jumping over or walking through others if they are allowed to move to cells behind the adjacent ones. Although the authors did not validate simulation outcomes with empirical data, the modelling ideas and the issues discussed in the paper remain an important contribution.

Blue et al. (1997) proposed a different approach that is based on transition rules. Agents move in the direction of the target and only deviate from this direction when conflicts with other, oncoming agents occur. If there is a conflict, certain rules determine the behaviour of agents. The author later extended this approach and also investigated the density-flow and density-speed relation in the simulation (Blue and Adler, 2001).

Inspired by car traffic models, Fukui and Ishibashi (1999) developed a similar approach. They considered cells in a row as lanes. Agents move in the direction of the lanes, and they only sidestep when the cell in front of them is occupied. If they have to change lanes, agents first try to evade diagonally to the front, and if that is not possible, they evade to the side. This decision rule is somewhat similar to part of the heuristics I describe in chapter 6 (section 6.3.1). Weifeng et al. (2003) simulated contra flow with the cells in the von Neumann neighbourhood as possible movement options for agents, that is, agents only step to the side with a certain probability but never move diagonally. Both approaches are based on simple local rules that describe the evasion behaviour of agents.

Muramatsu and Nagatani (2000) modelled the behaviour of pedestrians at a crossing with a biased random walk. In their first scenario, agents walk in two different directions and in the second scenario, agents walk in all four directions. This approach was inspired by the lattice gas model and is the first cellular automaton for pedestrian dynamics that is inherently probabilistic. The next position of agents is drawn from the possible movement options, which have transition probabilities that encode the system's behaviour. A similar model was later investigated by Tajima and Nagatani (2001).

Burstedde et al. (2001) presented another model that encodes behaviour in transition probabilities and introduced a concept they referred to as *floor field* to model long-range interactions. In addition to local transition probabilities, they use the analogy of chemotaxis. Agents follow a virtual trace others have left behind while passing a certain area. With the floor field, they implicitly introduced long-range interactions. They also discussed the difference between a static and dynamic floor field. The static floor field stays fixed over time, whereas the dynamic variant is affected by the agents and hence changes over time. Kretz et al. (2006b) and Kretz (2007) later extended the model with the possibility for agents to have different velocities.

In his doctoral thesis, Klüpfel (2003) described a cellular automaton with stochastic transition rules that can also degenerate into deterministic rules if the model parameters are chosen in a certain way. He proposed two methods for the navigation to the target. For the first method, the movement direction is assigned directly to each cell, which he suggested is the better option for evacuation scenarios because it requires less information to be stored (Klüpfel, 2003, p. 58). For the second method, the distance of each cell to the target is stored in the cell itself. The direction of the next cell chosen by agents is then computed as the direction of the gradient. He referred to the values stored in cells as *potential*. The notion of potentials is similar to the one in social force models. However, although agents implicitly chose positions with lower potential, the potential is not mathematically interpreted as such. If the potential was negated, it could be interpreted as utility and the motion process as utility optimisation (section 4.1, chapter 4).

Varas et al. (2007) presented a deterministic simulation that assigns a value to each cell based on the distance to the target. The algorithm they used to compute the values of this floor field is a variant of Dijkstra's algorithm (Dijkstra, 1959). Later, Klein et al. (2010) and Köster et al. (2011a) developed and calibrated another deterministic simulator that builds on the idea of a potential field. In their model, cells are hexagonal and positions are updated sequentially, which deviates from the original definition of cellular automata. The potential field includes repulsive contributions from nearby agents and obstacles to ensure that agents keep a certain distance to each other and to the obstacles. Leng et al. (2014) proposed an alternative cellular automaton with hexagonal cells and a floor field. Zhang et al. (2012) used a potential field, which they interpreted as cost for the trajectory to the target from the respective cells. Agents can be expected to choose the cell with the least cost, which, again, resembles utility optimisation. In contrast to utility optimisation, the next cell is chosen with stochastic transition rules.

Was et al. (2006) and Was and Lubas (2013) published a modification and extension of previous cellular automata. In their model, cells have relatively small side lengths of 0.25 m. Agents are placed at the centre of cells, while their physical representation is an ellipse that is oriented with the movement direction and reaches over the cell. They introduced a parameter that governs the accepted degree of overlap of two agents at a focal position. If the threshold is exceeded, the focal position cannot be reached by the agent. Was et al. (2006) used the concept of social space by Hall (1966), which describes several characteristic distances to simulate how pedestrians keep a certain distance to each other. In contrast to the original formulation of social space, Was et al. (2006) suggested that the characteristic distances are asymmetric, that is, affected by the orientation of pedestrians. Based on the idea of potentials, an increasingly repulsive force for closer positions around the focal agent is added. Using the cellular automaton with the social distance model, Was and Lubas (2014) later presented an agent-based simulation approach (section 3.4).

Zhang and Han (2011) described a cellular automaton with rectangular cells that is based on a floor field. They referred to the floor field as *potential field* and introduced mechanisms to reproduce follower behaviour. The decisions agents make is deterministic because they choose the adjacent cell in the Moore neighbourhood with the greatest value. With the simulation, they successfully reproduced lane formations and oscillations at bottlenecks. Feliciani and Nishinari (2016) used a floor field with stochastic transition rules and a sub-mesh to allow for densities of up to 10 pedestrians per square metre. To avoid unnecessary high densities in normal situations, agents are only allowed to enter the sub-mesh if they were unable to move forward for a certain time.

This section demonstrated that there are many different models that use a cellular structure. Some basic modelling choices stand out: the cellular automaton can be deterministic or probabilistic; it is based on a floor field or local interaction rules that determine the next step; the cells are rectangular or hexagonal. The cellular automata for pedestrian dynamics I discussed are grouped according to these categories in table 3.1. In addition to the basic set-up of the cellular automaton, many extensions and modifications can be made, which illustrates the flexibility of this approach.

Cellular automata have the advantage that they are computationally efficient. This is due to the inherent discrete structure of time and space in the simulation and the implicit spatial data structure of cells. The former allows for fast advancement of

	deterministic	probabilistic
local rules	Blue et al. (1997) Fukui and Ishibashi (1999) Blue and Adler (2001) Tajima and Nagatani (2001)	Muramatsu and Nagatani (2000) Weifeng et al. (2003)
floor field	Gipps and Marksjö (1985) Was et al. (2006) Varas et al. (2007) Klein et al. (2010) ⬡ Köster et al. (2011a) ⬡ Zhang and Han (2011) Was and Lubas (2013)	Burstedde et al. (2001) Klüpfel (2003) Kretz et al. (2006b) Kretz (2007) Zhang et al. (2012) Leng et al. (2014) ⬡ Feliciani and Nishinari (2016)

Table 3.1: Classification of cellular automata for the simulation of pedestrian flow. Publications marked with a hexagon (⬡) use hexagonal cells instead of rectangular cells. Deterministic models may also have some probabilistic aspect such as the choice between to cells that are equally attractive. In probabilistic models, on the other hand, the next cell is chosen based on stochastic transition rules. Local rules describe directly which cell is chosen by agents, for example, based on whether the cells ahead are free or not. In floor field models, agents evaluate the utility, cost, benefit, or potential of the field and choose an adjacent cell accordingly. Some of the models belong to more than one category or are not clearly attributable. In those cases, I assigned them to the category that seemed salient in the context of this work. The approaches assigned to the category *floor field* may not have been explicitly described that way by the authors, and the category may be debatable.

the simulation time because of the numerically large steps; the latter provides the states of neighbouring cells without search. Since no differential equations have to be solved, numerics are rather simple, which allows for fast implementation. The floor field concept is very flexible and can be extended with other behavioural aspects such as sub-group behaviour, which we carried over from an extension of the social force model (Moussaïd et al., 2010; Köster et al., 2011a). As a basic locomotion layer, cellular automata can be used for agent-based approaches (Was and Lubas, 2014).

There are certain limitations to the model. The fixed discretisation of space limits the study of microscopic motion and interactions in human crowds, such as stepping behaviour and positions in continuous space. The cell size usually also determines the physical extension of agents and hence overly limits the maximal density. Smaller cells may allow for more detail and show improvements in some phenomena. On the other hand, smaller cells may also lead to a loss in computational efficiency. The grid itself leads to movement artefacts that have to be dealt with in order to obtain unbiased motion in all directions (Gipps and Marksjö, 1985; Hartmann, 2010). Some of these issues can be overcome with the optimal steps model (Seitz and Köster, 2012), which is continuous in time and space but still allows for large discrete motion steps (chapter 4). Other aspects, such as continuous motion, inertia, and contact forces, are not observable in cellular automata. These aspects are reproducible by models described in the next sections, especially by velocity-based (section 3.2) and force-based models (section 3.3). Alternatively, a dedicated physical layer can be introduced to address such phenomena (chapter 5 in part II).

3.2 Velocity-based models

This section is dedicated to velocity-based simulation approaches. Velocity-based models are formulated in continuous space and time. For simulation in a computer, it has to be solved numerically and thus be discretised. In contrast to cellular automata, the next position is not chosen according to some rules or transition probabilities. Instead, the velocity is determined through a first-order ordinary differential equation. The velocity can then be numerically integrated to obtain the positions of agents at discrete simulation time steps. The basic equation for agent i is given by

$$\dot{x}_i = v_i, \tag{3.1}$$

where x_i is the position, $\dot{x}_i$ the first derivative of the position (the velocity), and v_i the function determining the velocity in the model. To solve a first-order ordinary differential equation, initial values for the vectors x and v representing the positions and velocities of all agents have to be provided at a known time point t_0. The resulting initial value problem is specified by

$$\begin{aligned} v(t_0) &= v_0, \\ x(t_0) &= x_0, \end{aligned} \tag{3.2}$$

where v_0 and x_0 are vectors containing the initial velocities and positions of all agents.

Approaches that have been studied in car traffic simulation for some time are optimal-velocity models (Newell, 1961; Whitham, 1990; Nakanishi, 2000; Tordeux and

Seyfried, 2014), which are also referred to as *car-following* models. Car following was first studied empirically and theoretically by Chandler et al. (1958) and has since inspired model development (Gazis, 2002). In optimal-velocity models, the velocity of car i is determined through the (optimal) velocity function $V(\Delta x_i)$ given the distance $\Delta x_i = x_{i+1} - x_i$ to the next car $i+1$ ahead. With the velocity function $V(\Delta x_i)$, the speed can be determined directly:

$$\dot{x}_i(t) = V(\Delta x_i(t)). \tag{3.3}$$

To account for reaction times, a relaxation time $\tau > 0$ can be introduced (Newell, 1961; Tordeux and Seyfried, 2014), which results in the more complex system

$$\dot{x}_i(t+\tau) = V(\Delta x_i(t)). \tag{3.4}$$

In real physical systems, the speed does not change arbitrarily but only through gradual acceleration. This can be simulated with a second-order equation of the form (Bando et al., 1995; Sugiyama, 1999; Mitarai and Nakanishi, 1999; Tordeux and Seyfried, 2014):

$$\ddot{x}_i(t) = \frac{1}{\tau}[V(\Delta x_i(t)) - \dot{x}_i(t)]. \tag{3.5}$$

Although still an optimal-velocity model, the resulting system has second order and, therefore, does not belong to the velocity-based models but to the force-based models (section 3.3). Tordeux et al. (2015) presented an optimal-velocity model for pedestrian dynamics.

In robotics, collision avoidance is a common problem (Kant and Zucker, 1986; Fraichard, 1993), and it is often important that the chosen trajectory be efficient, which is why it has to be optimised in some way. Fiorini and Shiller (1993) proposed to calculate the relative velocity of an obstacle – which can be another agent – to the focal agent. Given the relative velocity, a collision sector is computed. The collision sector contains all velocity vectors that lead to a collision. This cone-shaped area can be transformed back into the space of absolute velocities, and a velocity that does not lead to a collision can be chosen outside of it. The computation only takes into account the current velocities. It does not account for changes in speed and direction of motion. Fiorini and Shiller (1993) excluded velocities that lead to a collision after a certain threshold to facilitate computation and account for the limited credibility of long-term predictions. They referred to the resulting area as *velocity obstacle*, which describes that velocities chosen in this area would lead to a collision.

Using the velocity obstacle method, Fiorini and Shiller (1998) later proposed an approach for motion planning in robotics. Given the set of velocities that lead to collisions, an agent can select a variety of velocities outside of that area that does *not* lead to collisions. Shiller et al. (2001) extended the velocity-obstacle method for non-linear motion, and Berg et al. (2011) developed an algorithm that takes into account the reaction of other agents. The approach has since been used in robotics to plan motion and in animation to simulate the behaviour of agents in virtual environments (van den Berg et al., 2008; Curtis, 2013). Curtis and Manocha (2014) used an extension of the velocity-obstacle approach to compute a density-speed diagram, which was an

optimal velocity (car traffic simulations)	Newell (1961) Whitham (1990) Nakanishi (2000) Tordeux and Seyfried (2014)
obstacle velocity	Fiorini and Shiller (1998) Shiller et al. (2001) van den Berg et al. (2008) Curtis (2013) Curtis and Manocha (2014)
gradient navigation	Dietrich and Köster (2014) Dietrich et al. (2015)

Table 3.2: Classification of velocity-based models. The optimal-velocity models listed were developed for car traffic simulations and hence are one-dimensional. Obstacle-velocity models were mainly introduced for robotics (Fiorini and Shiller, 1998) and only later carried over to animation and computational science (Curtis and Manocha, 2014).

important step for this model in the direction of pedestrian stream simulation in scientific computing.

The gradient navigation model (Dietrich and Köster, 2014) is another velocity-based model. Here, the motion direction of agents is determined through the direction of a gradient on a continuous navigation field. The navigation field can be interpreted as utility, travel cost, or potential at the respective positions in the plane. The speed is determined through a second-order equation that contains a relaxation constant, which also introduces acceleration into the model. Therefore, it is not purely velocity-based but also has aspects of a force-based model. It has been used to investigate the occurrence of stop-and-go waves in a one-dimensional system with periodic boundary conditions (Dietrich et al., 2015). The authors found that there is a clear threshold in the number of agents that determines whether stop-and-go waves occur or not in the gradient navigation model.

An important advantage of velocity-based models is that they are usually continuous in space and time. Nevertheless, they could also be evaluated at coarse time steps, which would make them discrete in time and space. For example, the gradient navigation model can be discretised in a way that it yields a discrete steps model similar to the optimal steps model (Seitz et al., 2016). The formulation of movement in closed mathematical equations can be advantageous for analysis and reveal interesting features of the model (Dietrich et al., 2015). Using scalar fields – as in the gradient navigation model – provides modelling flexibility and allows for the introduction of mechanisms used in cellular automata and the social force model. Not computing forces but determining the velocity directly has the advantage that potentially undesired effects such as inertia do not occur.

In velocity-based models, the decision making of agents is encoded in the velocity function. This poses a certain limitation for modelling advanced behavioural aspects. It can also be questioned whether human movement decision making can be represented with such equations in general. For the simulation of the physical environment, the velocity function is also limited because physical interactions are typically modelled with forces, which lead to a second-order differential equation. Nevertheless, due to their conciseness, they may provide a parsimonious alternative to other approaches. The models I discussed in this section are summarised in table 3.2.

3.3 Force-based models

The underlying idea of force-based models is "that a pedestrian acts *as if* he/she would be subject to external forces" (Helbing and Molnár, 1995). The statement means that pedestrians follow attractive potentials and avoid repulsive potentials that represent their motivation to behave in a certain way. Helbing (1991) described it as "some kind of psychic tension, which causes the individual to act toward its aim in order to diminish this tension". These motivations or tensions are induced by the environment such that they could be seen as external (physical) forces. However, it is a model of decision making and social behaviour, not physics. This can lead to misunderstandings across disciplines, especially, since the social force model can be combined with physical contact forces.

The idea of social forces was originally introduced by social psychologist Lewin (1951), who was also referenced by Helbing and Molnár (1995) in their publication introducing the social force model for pedestrian dynamics. The idea of forces guiding pedestrian motion was described before the social force model had been published as such:

> The model hypothesizes the existence of repulsive forces between pedestrians so that as the subject approaches another pedestrian the 'potential energy' of his position rises and the 'kinetic energy' of his speed drops. The repulsive forces also deflect him from a straight line. The situation is loosely analogous to that of a body moving through gravitational fields generated by a number of other objects, except that the forces are repulsive rather than attractive, and not necessarily symmetric about the bodies concerned. (Gipps and Marksjö, 1985)

Although Gipps and Marksjö (1985) did not implement this concept in differential equations representing physical forces, the idea behind it seems like the social force model. Reynolds (1987) used a force field to simulate flocks of birds and bird-like objects (which they referred to as *boids*):

> The force field model postulates a field of repulsion force emanating from the obstacle out into space; the boids are increasingly repulsed as they get closer to the obstacle. This scheme is easy to model; the geometry of the field is usually fairly simple and so an avoidance acceleration can be directly calculated from the field equation. (Reynolds, 1987)

In this case, the forces are actually translated into physical forces and motion is computed accordingly. In car traffic simulation, some optimal velocity models are also based on a second-order differential equation (Bando et al., 1995; Sugiyama, 1999; Mitarai and Nakanishi, 1999; Tordeux and Seyfried, 2014). However, these systems are one-dimensional in principle (section 3.2).

The idea of social forces can be formulated mathematically (Helbing, 1993; Helbing and Molnár, 1995; Molnár, 1996), which leads to equations well-known in physics:

$$m_i \times \ddot{x}_i = f_i, \tag{3.6}$$

where m_i is the mass, $\ddot{x}_i$ the acceleration, and f_i the forces acting on agent i. The equations more naturally describe physical behaviour such as granular flow (Rao and Nott, 2008). Nevertheless, pedestrians are also subject to physical forces and acceleration.

The current position x_i is two-dimensional and so are the forces in f_i, which are described as vectors. The direction of f_i is the direction of acceleration, and its length $||f_i||$ represents the strength of the acceleration. The current velocity of particle i is $\dot{x}_i$. In a dynamic simulation model, the position x_i, the velocity $\dot{x}_i$, the acceleration $\ddot{x}_i$, and the forces f_i can change over time. Thus, they are parametrised by the time t, such that

$$m_i(t) \times \ddot{x}_i(t) = f_i(t). \tag{3.7}$$

This formulation is a second-order ordinary differential equation, which can be solved numerically to obtain the current position of the agent. As for velocity-based models (section 3.2), initial values at t_0 for the positions x and velocities $v = \dot{x}$ must be specified to solve the system of equations.

On the right side of equation (3.7), one still has to define the acting forces. In granular flow, these are contact forces, gravitation, and so on. For human behaviour, these are the motivations such as the need to keep a certain distance from walls or other individuals or the desire to reach a target. Helbing and Molnár (1995) formulated f_i as the sum of the different motivations in a way that the resulting force f_i is composed of all these influences:

$$f_i = f_t + \sum_j f_{p,j} + \sum_k f_{o,k}, \tag{3.8}$$

where f_t is the attractive force toward the target, $f_{p,j}$ the repulsion induced by another pedestrian j, and $f_{o,k}$ the repulsion from obstacle k. Helbing and Molnár (1995) additionally introduced terms for the attraction to others agents such as friends or family members and a term for fluctuations – both of which I do not consider here for simplicity.

Calibration of the model is carried out through the choice of functions and their parameters within f_i. Advanced aspect of pedestrian behaviour such as small group coherence can be introduced by adding further terms (Moussaïd et al., 2010). Therefore, the behaviour of pedestrians is indirectly described as the result of forces and finally acceleration in tow-dimensional space. Computing the acceleration given the current state of the system results in a dynamical simulation and can finally be interpreted as the motion of pedestrians.

The social forces conceptionally do not mean that these forces physically exist but are then used in a physical model to accelerate the pedestrians as if they did. Since the forces are used within a physical model to move agents, they can easily be combined with actual physical forces such as contact forces. Helbing et al. (2000a) introduced contact forces to the social force model to simulate dangerous crowd situations with high densities. They called the observed behaviour “escape panic”. However, there is considerable doubt that "panic" is the right description for human behaviour at mass events that led to disasters (Johnson, 1987; Aguirre, 2005; Drury et al., 2013).

Contact forces are well known from granular flow and the discrete element method (DEM, Cundall and Strack, 1979). The difference to molecular dynamics is that the forces only become active if the particles touch. In the case of circles, this means that the sum of their radii is smaller than the distance between their centres. The DEM also describes the rotation of a particle with Euler's equations (Kleinert et al., 2013). Mathematically, the social force model without contact forces could be compared to molecular dynamics and with contact forces, to a combination of molecular dynamics and granular flow.

Helbing et al. (2000a) used one circle for the representation of an agent's body. If the bodies are represented by contact forces, this is a limitation of the model because the shape may influence the flow and clogging behaviour. Therefore, Langston et al. (2006) represented the human body with three circles. Two circles form the shoulders and another, bigger one the pedestrian's torso. They used this model to simulate contra-flow (Smith et al., 2009) and sub-group behaviour (Singh et al., 2009). Chraibi et al. (2010) used elliptic shapes for the body to simulate flow in a corridor and considered the swaying of pedestrians while they walk. In centrifugal force models (Yu et al., 2005; Chraibi et al., 2010), the velocities and not only the positions of other agents are taken into account. Agents in front are not repulsive to other agents walking behind them if they walk in the same direction with the same speed. When the agent in front stops, the repulsive forces effect the following individuals again.

The social force model and its derivatives show a series of phenomena, most of which are qualitative. In their original publication, Helbing and Molnár (1995) described lane formation in contra-flow simulations. Agents form lanes when they walk in opposite directions in a corridor. Additionally, they found oscillations in a scenario where two pedestrian flows pass through a bottleneck in opposite directions. After one individual has passed through the door, a whole group follows in the same direction until the pressure on the other side reaches a certain strength. Yu et al. (2005) also observed lane formation in the centrifugal force model.

Arching and clogging at bottlenecks, which is suddenly released in an avalanche, was later described for an extended model (Helbing et al., 2000a). This phenomenon seems to be present in the simulations because of the analogy to physics as arching and clogging is a phenomenon well-known in granular flow (Pöschel, 1994; To et al., 2001). The authors demonstrated the faster-is-slower effect because of clogging: when the crowd moves faster, clogging can occur, which leads to slower egress. There is some evidence supporting the faster-is-slower effect in recent experiments with sheep and students (Garcimartín et al., 2014; Pastor et al., 2015). Clogging was also found for the centrifugal force model (Yu et al., 2005). The faster-is-slower effect was further investigated for social force models by several authors (Lakoba et al., 2005; Parisi and Dorso, 2007).

Helbing et al. (2000a) introduced a "panic parameter", which controls whether individuals follow the average direction of others in a certain proximity or their individual direction when searching for an exit they cannot see. The behaviour of following others was called "herding behaviour" by the same authors. Other analogies from physics have been investigated for the social force model, such as "freezing by heating" (Helbing et al., 2001). This phenomenon stems from particle physics (Helbing et al., 2000b) and describes how more fluctuations can result in a crystallised state. In the context of pedestrian dynamics, they interpreted this as individuals "panicking".

Another noteworthy extension to the original model is the concept of sub-group behaviour with sub-groups of up to four members. Moussaïd et al. (2010) described how the desire to communicate with each other and at the same time turn the head as little as possible lets small pedestrian groups walk abreast. Based on this idea, the authors introduced additional forces that lead to this emergent behaviour. They used video observations for their work and also studied the distribution of the number of pedestrians usually found in a sub-group.

After the initial development of the social force model, a line of research has focused on the calibration of parameters and the validation of simulation outcomes (Helbing et al., 2005; Johansson et al., 2007; Moussaïd et al., 2009b; Parisi et al., 2009). Some numerical issues were identified, and a mollified version was suggested by Köster et al. (2013) as a solution. Although computing force-based models is certainly possible, it poses a greater numerical challenge than cellular automata.

I argue that there is a conceptional difficulty in social force models. At first, the forces are understood as psychological tension or motivation and then used to describe physical motion. In the end, the model's behaviour is validated through its outcome, that is, the crowd's movement. However, they lack a meaningful interpretation as a psychological process: social force models neglect the transition from decision making to actual pedestrian motion because they translate the psychological tension *directly* to physical motion. This could be resolved by simply regarding social forces as a model of the observable motion of pedestrians and crowds. Otherwise, it seems unclear what is part of the decision-making layer and what is part of the locomotion layer.

While force-based approaches allow for the introduction of physical forces, such as contact forces between agents, they can also entail other emergent effects that are not necessarily realistic (Chraibi et al., 2011). For example, inertia is certainly a realistic effect in physics, but social forces also lead to inertia, which is not a desirable effect. This can be observed in simulations where agents bounce off the walls or pass over the target. The concept of forces acting on pedestrians may also be questioned on a conceptual level. The model is clearly inspired by physical systems such as particles or grains. While appealing at first, the analogy seems to be misleading from a psychological perspective. Pedestrians (like animals and robots) are self-driven objects and actively move in the environment or even actively shape it. This fact is not well-captured by the conceptual modelling idea. Nevertheless, the social force model has popularised the study of pedestrian dynamics in computational science and has produced a number of testable hypotheses.

3.4 Alternative approaches

In this section, I describe several approaches that do not belong to one of the other three categories discussed in the previous sections. Some of the models in this section have aspects of the previous categories or are even partly based on them. I chose to allocate them to this section when they depart from the original modelling idea as a principal concept. The main branches are models based on steering behaviours, agent-based models, and what I refer to as *cognitive approaches* to modelling pedestrian behaviour.

Reeves (1983) described particle systems for animation and proposed that individual particles can be moved by "adding its velocity vector to its position vector". This seems like a velocity-based approach, but he also suggested that, for more complex animations, acceleration can be used, which would mean it is a force-based approach. It is not entirely clear from the paper itself how the velocity and the acceleration are determined. It seems as if they are rather constants and hence are not modelled other than by assigning a value once.

Based on particle systems, Reynolds (1987) proposed an animation approach for flocks of birds and bird-like flocks, which he called *boids*. He described three rules that govern the behaviour of the simulated objects. Reynolds (1987) let behaviour affect the motion process via forces, and hence, it is a force-based approach. However, changes in direction and speed could also be directly integrated into the velocity.

Later, Reynolds (1999) described steering behaviours. The idea is similar to the one presented in his previous paper on flocks: steering behaviours control locomotion but are not directly translated into it. Steering and locomotion are, in principle, independent. According to his publication, the locomotion layer was realised with a force-based approach. Ondřej et al. (2010) built on the concept of steering behaviours but based the mechanisms for collision avoidance on findings in cognitive sciences, namely on the findings from Cutting et al. (1995).

The next category are agent-based models. This category does not seem to be well-defined as its descriptions are usually rather vague and do not clearly demarcate it from other approaches. I use a definition and classify models as agent-based if they show some typical features of it or if the authors put special emphasis on the agent-based approach. Bonabeau (2002) and Goldstone and Janssen (2005) published general perspectives on agent-based modelling. The following definition is intended to give a starting point for the discussion:

> In agent-based modeling (ABM), a system is modeled as a collection of autonomous decision-making entities called agents. Each agent individually assesses its situation and makes decisions on the basis of a set of rules. Agents may execute various behaviors appropriate for the system they represent—for example, producing, consuming, or selling. (Bonabeau, 2002)

This is a broad definition and is also met by the models discussed before. The crucial aspect I want to stress is the focus on individual assessment and decision making. For this, agents typically have individual attributes such as physical features and personal goals.

A good example of an extensive agent-based model is the computer game "Cities: Skylines" (Paradox Interactive, 2015), in which gamers build a city. For the simulation of the built city, citizens (the agents in the game) have individual representations with a name, age, home, and other attributes. The citizens go to work and interact in an extensive traffic system (Lehto et al., 2015). This stresses the agent-based approach: individuals in the simulation have different attributes that lead to a variety of behaviours and interactions. The differences go beyond parameters such as preferred speed. Nevertheless, attributes like preferred speed, body size, and weight could still be considered *aspects* of agent-based modelling.

Feinberg and Johnson (1995) proposed a simulation for the evacuation response after a fire alarm. They equipped agents with individual information such as social ties, the entrance they used, and a perception score, which determines how likely agents are to evacuate quickly. Zarboutis and Marmaras (2004) simulated a tunnel fire in a subway. Apart from the simulation of fire, smoke, and the technical system, they introduced two groups of agents with different features: passenger and the metro personnel. The behaviour of the personnel was not included in the simulation, and only for the passenger, concrete behavioural rules were given.

Chu et al. (2011) proposed a model that separates sensors, the cognitive system, and actuators – a representation of agents that is common in artificial intelligence (section 6.2 in chapter 6). The agents have specific social roles and functions, which the authors use to simulate social behaviour in emergency situations. This approach seems to be strongly influenced by artificial intelligence (Russell and Norvig, 2010, p. 35), but the authors also tried to bring in background form social sciences. A similar model was reported by Chu and Law (2013), who separated individual, group, and crowd behaviours. They introduced an "individual experience profile", group membership, and social traits. Schneider (2011) aimed to study "human panic behaviour" with simulation. In his doctoral thesis, he developed an extensive simulation with mechanisms that determine the behaviour of agents. He did not, however, validate the predictions or compare them to empirical data.

The bombings from 2005 in the London underground were simulated by von Sivers et al. (2014) using the Vadere simulation framework and the optimal steps model as a basis for decision making and locomotion. The behaviour that supposedly led to helping strangers in the real events is based on the social identity approach (section 7.1, chapter 7). Building on this work, von Sivers et al. (2016) studied the model's behaviour quantitatively in order to obtain insights into how helping behaviour affects evacuation times in such events. Due to the different goals and behaviours represented in the simulation, it can also be considered an agent-based approach.

Dijkstra et al. (2001) and Dijkstra et al. (2006) developed an agent-based simulation with a cellular automaton for locomotion. Was and Lubas (2014) used a cellular automaton with the social distance model (section 3.1), proposed an agent-based simulation framework, and compared competitive and non-competitive behaviours. Also using a cellular automaton, we introduced a model for the coherence of small groups (Seitz et al., 2014a) and studied the model's behaviour by comparing simulation outcomes to a controlled experiment with students (Köster et al., 2011b). Furthermore, we used mechanisms for the separation and reunion of small groups (Seitz et al., 2011), which allows for a more general application of the model in different scenarios (section 7.2, chapter 7). As these mechanisms create heterogeneous agents and behaviours, they can be considered agent-based.

Agent-based modelling allows for the introduction of individual features and behaviours and thus provides great flexibility. At the same time, this is a challenge for validation. The behaviours are not always based on modern, evidence supported background from cognitive sciences, psychology, or other behavioural sciences. The combination of a variety of mechanisms makes it intractable to thoroughly test the model's behaviour and compare it to empirical data. For some applications such as computer games and animation this may not be an issue. In computational science, however, the objective is to provide testable hypotheses through simulation. If the underlying model is too complex and versatile, this becomes an increasingly challenging task. Nevertheless, the idea of individual differences in agents and the technical modelling concepts for it may provide useful tools for scientific model development.

Hoogendoorn and Bovy (2003) proposed to represent walking as a differential game. Agents predict the behaviour of others by taking into account their current observation. They later suggested that pedestrians minimise the cost of walking and divided the model in a physical part and control part (Hoogendoorn, 2007). Guy et al. (2010) and Guy et al. (2012) also presented a model that lets agents minimise effort to find a collision-free trajectory.

Antonini (2005) and Antonini et al. (2006) proposed a discrete choice model for pedestrian simulation. In discrete choice models, "each alternative in a choice experiment can be associated with a latent quantity, called utility. The alternative with the highest utility is selected" (Antonini et al., 2006). This idea is especially relevant for the optimal steps model (discussed in the next chapter) because it makes a similar assumption. In contrast to the optimal steps model, the utility is a random variable. Agents choose from cells around them that correspond to different directions and speeds. Therefore, there is a clear difference between this approach and the optimal steps model, yet there are also some interesting parallels in basic modelling ideas, specifically the notion of utility optimisation.

Pellegrini et al. (2009) described a model in which they assumed that all agents know the positions and velocities of all other agents around them. With this information, the agents predict the movement of other agents. The authors also showed how to train the model's behaviour with video recordings of real pedestrians. Later, Pellegrini et al. (2010) extended the model with stochastic decision making for ambiguous situations. The assumption about the knowledge of all positions and velocities at the same time seems to exceed realistic cognitive capacities of humans.

Moussaïd et al. (2011) proposed that pedestrians maximise the time to collision and chose their directions and speeds accordingly. For this, agents predict the motion of others and take into account their future positions. In their publication, the authors referred to this as *simple rules* that determine pedestrian behaviour. The rules are mathematically simple, but the computation can be costly because of the rather complex optimisation problem. It could also be questioned whether optimisation is a plausible model for human cognition in general (subsection 6.1.3, chapter 6). Another interesting aspect of this approach is that the decision is passed on to a physical layer that realises the locomotion. The physical layer is basically an adaptation of the social force model, which allows for the simulation of crowd turbulences and other phenomena that are based on physical interactions.

Animation of avatars and crowds is a topic researched in computer science. I discussed some of these approaches that seemed especially relevant for computational science. For a more thorough treatment of this field, I refer to the textbooks by Pelechano et al. (2008) and Thalmann and Musse (2012). Individual-based modelling is also common in animal behaviour (Grimm and Railsback, 2005). Especially collective animal behaviour (Sumpter, 2006) is an active research field that can give insights into human behaviour and provide useful methodology and models for the study of pedestrian flow. For example, random walk models describe individual motion as a (biased) random walk in the spatial domain (Codling et al., 2008), which is a common approach to simulating individual animal movement. Vicsek and Zafeiris (2012) published a review on collective motion in animals.

The great variety of approaches makes it difficult to decide which are relevant for a specific purpose. Some aim at the introduction of findings from social and cognitive sciences, others do not seem to be based on such criteria or even explicitly do not aim at them. If models are not investigated by many authors, it is also challenging to judge their validity for scientific purposes in general. On the other hand, there is a wealth of ideas, methods, and phenomena that can inspire model development across disciplines, which makes it worth looking at other fields.

3.5 Similarities and differences

I investigate the conceptional similarities and differences of pedestrian stream simulation models in this section. Some have been made explicit already in the previous sections and are revisited here. I mainly focus on the perspective of scalar fields (Seitz et al., 2016), which allows for a common mathematical basis for many of the models[2]. Identifying similarities and differences is important to select a suitable model or develop a new approach if no other model meets the requirements. I argue that models based on scalar fields can be efficient for practical application but lack a plausible representation of the natural locomotion and decision-making process.

In the previous sections, I also discussed models of car traffic. These approaches have established cellular automata and optimal-velocity models for traffic simulation before they became a focus in pedestrian dynamics. This may be reasonable because of the similarity in phenomena of interest, such as the density-speed relation and stop-and-go waves. However, car traffic is mainly studied as a one-dimensional system that may include some lane-changing mechanisms. Some early models of pedestrian dynamics used a similar methodology (e.g., Fukui and Ishibashi, 1999), but in general, pedestrian dynamics are better captured with two-dimensional systems. Some pedestrian phenomena may still be investigated in one-dimensional systems (e.g., Dietrich et al., 2015).

Another major difference between car traffic and pedestrian dynamics is the physical system, which constrains locomotion and behaviour. Cars can be described and are understood well in physical terms because it is a human-built machine. Pedestrian physics and biomechanics, on the other hand, is more complex and not that well understood (Winter, 2009). For example, no bipedal walking robot to date can walk as skilful as a human (Buschmann et al., 2015). Pedestrians have more freedom in their motion behaviour than drivers and pedestrian traffic is usually less regulated and

[2] We developed the perspective of scalar fields and the superposition principle in a collaborative paper (Seitz et al., 2016). The ideas used in this section are largely based on this publication.

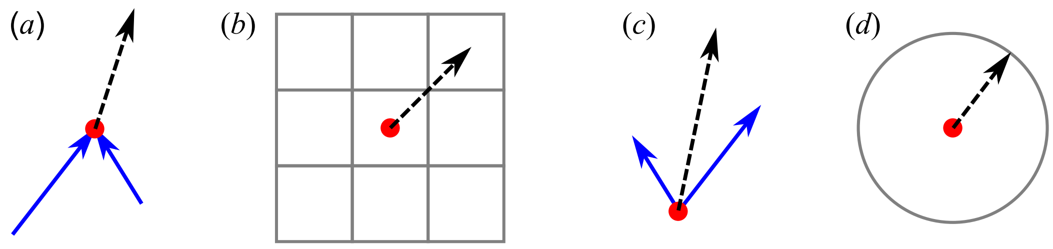

Figure 3.2: Illustration of four microscopic pedestrian stream simulations: force-based models (a), cellular automata (b), the gradient navigation model (c), and the optimal steps model (d). All of them usually employ scalar fields for the decision-making process, but the interpretation of the scalar fields varies across the models. (Figure: Seitz et al., 2016)

two-dimensional, which leads to more complex interactions. I give more information on the biomechanics of walking in chapter 5, section 5.1. Walking has come a long way in evolution (Hunt, 1994) and is the original form of human locomotion. Therefore, human cognition may be more adapted to walking than to driving. Finally, social norms may have a greater influence on pedestrian behaviour than in car traffic and lead to variations across cultures (e.g., Chattaraj et al., 2009).

There are certain commonalities between systems of car traffic and pedestrian dynamics, but there are also major differences. It may be reasonable to start with a one-dimensional system that was adopted from car traffic. For a comprehensive simulation model that allows predictions of practical scenarios, a two-dimensional system is necessary. Extending car traffic simulations with two dimensions may provide a good basis but has other conceptional limitations. Perhaps the similarities could be better addressed if the physical layer was separated clearly from the decision-making layer. Then, a car traffic simulation could benefit from findings in pedestrian research on the decision-making layer and vice versa.

The major categories in pedestrian stream simulation – which are also represented in the structure of the previous sections – are force-based models, velocity-based models, and cellular automata (figure 3.2). They have proven to successfully reproduce various phenomena. Alternative approaches not based on these concepts or extending them have been developed and contain interesting ideas and mechanisms. Apart from this classification, some early cellular automata stand out because agents make decisions based on simple rules (Blue et al., 1997; Fukui and Ishibashi, 1999). Although these models are based on a cellular grid, they are conceptionally different to approaches with scalar fields.

In the following paragraphs, I discuss the perspective of scalar fields and the superposition principle, which we proposed in a paper dedicated to this topic (Seitz et al., 2016). The underlying idea of this perspective is that pedestrian motion is guided by attractive and repulsive influences from the social and built environment. Specifically, the target is attractive, and obstacles and other pedestrians are repulsive.

It is a simple model of behaviour that can be interpreted as based on approach and avoidance motivation (Elliot, 2006). The concept is very flexible, and many of the previously discussed models build on it, although often without explicitly stating it.

The concrete models seem to be very different at first. The idea of attractive and repulsive forces had been described before the social force model was proposed (section 3.3), which shows that there are certain similarities in the basic modelling concepts. Local repulsion and attraction are described by a distance function. In most cases, this function is symmetric around other agents, obstacles, or the target. Either attraction has negative values and repulsion positive values or the other way around. Which one of the two is positive depends on the interpretation of the values. If it is interpreted as utility or benefit, attractive contributions are positive and repulsive ones negative. In the case of potential or cost, it is the other way around.

Independently of the interpretation, the individual contributions of elements in the environment are usually superposed (summed up). Then, each position in the plane has a scalar value, which is defined by the function

$$\begin{array}{rlcl} s: & (\mathbb{R}^2, \Theta) & \to & \mathbb{R} \\ & (x, \vartheta) & \mapsto & s(x, \vartheta), \end{array} \tag{3.9}$$

where Θ is the space of possible states, $\vartheta \in \Theta$ the current state, and $x \in \mathbb{R}^2$ the position in the plane that is being evaluated. As the output of the function s has only one value in $\mathbb{R}$, it is a scalar function. Since s is available for all positions x of the simulated environment, the function defines a scalar field that can be used for a model of pedestrian dynamics. The scalar field may be interpreted as utility or potential.

Some models are not based only on a distance function but a more complex algorithm. This is the case with floor fields (Burstedde et al., 2001) and navigation fields (Kirik et al., 2009; Kretz, 2009; Hartmann, 2010; Kretz et al., 2011; Hartmann and Hasel, 2014). In the former case, the scalar field can be influenced by the previous trajectories of other agents; in the latter case, the target attraction can be computed by a propagating wave front. Computing the arrival time of the wave front is numerically rather complex (Sethian, 1999, 1996), but the result can still be stored in a grid. Through interpolation between grid points, a value for arbitrary points in the plane is obtained (Hartmann, 2010; Seitz and Köster, 2012). More details on this topic can be found in chapter 4, subsection 4.1.2.

Models that use the scalar field approach are cellular automata with floor fields, social force models, the gradient navigation model, and the optimal steps model. The interpretation of the scalar field and how it is used for decision making and finally locomotion varies. In cellular automata, it is interpreted as either potential, benefit, or probability. The potential is not translated into forces. Instead, the cell with the smallest value is chosen, which can be seen as utility, cost, or benefit optimisation. When interpreted as probability, the values of the reachable cells have to be normalised. Then, they represent a discrete probability distribution from which the next position (or cell) is chosen. In the social force model, the scalar field is interpreted as a potential field, which leads to forces. Usually, not the potential field but the forces are constructed directly, which means they use a vector field. Nevertheless, the underlying potential is a scalar field. The same is the case in the gradient navigation

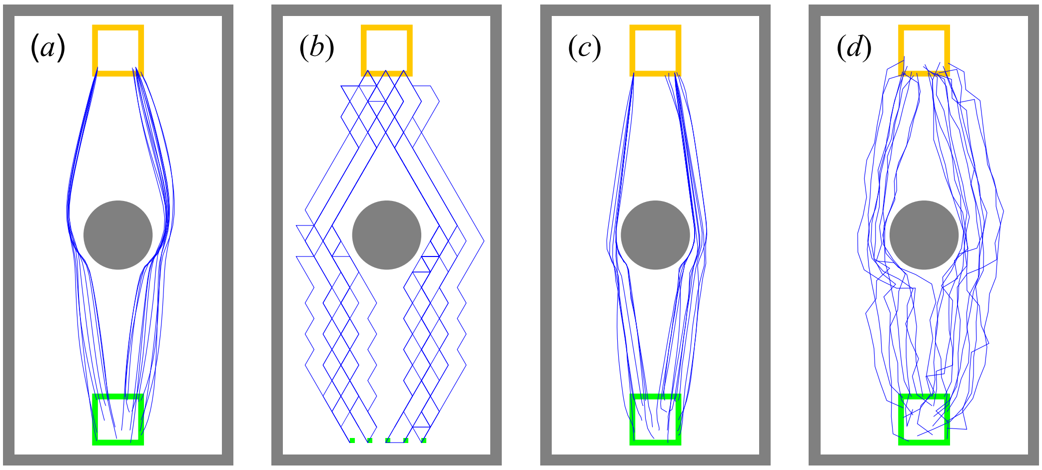

Figure 3.3: Emergent effects in a simulation scenario with four simulation approaches based on scalar fields: a social force model (a), a cellular automaton (b), the gradient navigation model (c), and the optimal steps model (d). The blue lines show the trajectories of agents. Agents are created on the bottom within the green areas (for the cellular automaton, agents are created on the grid cells). They walk around the (grey) column in the middle to the yellow target on top. The social force model and the gradient navigation model both have a continuous motion process and hence produce smooth trajectories. The cellular automaton shows the possible motion steps on the grid. With the optimal steps model, arbitrary positions are possible, but steps are discrete. (Figure: Seitz et al., 2016)

model where usually the gradient is constructed directly and is not calculated via the derivative of the scalar field. In the optimal steps model, the scalar field is best interpreted as utility, although we referred to it as *potential* in the original publication (Seitz and Köster, 2012).

The different use of the scalar field produces distinct emergent effects. We presented a paper on this in which we show how motion patterns differ between the social force model, the gradient navigation model, a cellular automaton, and the optimal steps model (Dietrich et al., 2014). In figure 3.3, the motion of individual pedestrians around a column is shown. It can be seen that a discrete locomotion process also produces coarser trajectories. The social force model and the gradient navigation model are continuous in time and space and thus yield smooth trajectories. The possible movement steps in cellular automata are clearly visible. In the optimal steps model, trajectories are coarser than in the models with a continuous motion process, but the motion of agents and their positions is not bound to a grid.

Given the same scalar field, it can be shown that the deterministic cellular automaton, the optimal steps model, and the gradient navigation model differ mainly in their discretisation (Seitz et al., 2016). I discuss the convergence of the optimal steps model and the cellular automaton in chapter 4, section 4.2; the convergence of the optimal steps model and the gradient navigation model is outlined briefly here.

In the gradient navigation model, the direction of the next step is calculated through the gradient of the scalar field. This can be understood as local optimisation with gradient descent. In the optimal steps model, the position of the next step is optimised directly on the step circle around the current position, which represents explicit local optimisation. If the direction of the gradient is used in the optimal steps model, only the discretisation and velocity are different. Hence, increasing motion steps in the gradient navigation model lets it converge towards the optimal steps model. We referred to the model with discrete steps as in the optimal steps model but with the direction from the gradient navigation model as *gradient steps model* (Seitz et al., 2016). The convergence illustrates the similarities among the models.

The concept of scalar fields is powerful but also has its limitations (Seitz et al., 2016). For example, the superposition of binary interactions can be questioned (Moussaïd et al., 2011). The calibration of model parameters is often carried out for specific scenarios, which makes it difficult to find a general model that reflects the behaviour in a variety of situations. Therefore, the extensibility is limited with the superposition principle. Concepts developed for one model can be carried over easily to another model based on scalar fields. For example, we developed a model for small group coherence in a cellular automaton (Köster et al., 2011b; Seitz et al., 2014a), which is based on a model originally proposed for the social force model (Moussaïd et al., 2010).

Simulations that are not based on scalar fields, such as the cellular automata with simple rules and steering behaviours, have their own challenges. The main reason I propose to use another decision making concept in part II is the limited flexibility and that the scalar field cannot be considered a plausible model of human cognition. It is, however, an approach that can produce certain emergent effects and phenomena and has inspired a wealth of research into simulation and model development for pedestrian dynamics. Therefore, the suitability of scalar field models depends on the application and phenomenon of pedestrian dynamics it aims to describe. As a description of the decision-making process, the models are limited.

3.6 Summary

In this chapter, I reviewed the literature on microscopic pedestrian stream simulation. I focused on modelling methodology rather than mathematical analysis or emergent effects and phenomena. The major categories that are reflected in the structure of the first sections are cellular automata, velocity-based models, force-based models, and other approaches. In the last section, I discussed similarities and differences – mainly with the perspective of scalar fields. Car traffic models seem to be influential in the historical context of pedestrian stream simulation, but these systems are also fundamentally different by nature.

Cellular automata (section 3.1) are mainly characterised by the discretisation of space into cells. Originally, the parallel update and the locality of rules were required. Some cellular automata for pedestrian dynamics go beyond this definition. Cellular automata can be classified based on some characteristic features: deterministic versus probabilistic, rule-based versus based on a floor field, and rectangular cells versus hexagonal cells. Many extensions and modifications have been proposed to capture additional phenomena. The greatest advantage is their simplicity, which both allows for fast implementation and computation. Their greatest limitation is the coarse discretisation of space, which leads to certain artefacts that have to be dealt with.

Velocity-based models (section 3.2) have been mainly used in car traffic simulations. They are characterised by the explicit modelling of the current velocity. This formulation leads to an ordinary differential equation of first order. Some models categorised as alternative approach also model the velocity but do not formulate it directly with an equation. The gradient navigation model can be seen as velocity-based model that has been used for the simulation of pedestrian dynamics. The definition of motion in a velocity equations is concise but may also lack both the advantages of a force-based process on the locomotion layer and the plausibility of a cognitive process on the psychological layer.

Force-based models (section 3.3) have been probably most influential in the scientific debate on pedestrian dynamics. Here, the force acting on agents is explicitly modelled, which leads to a second-order ordinary differential equation that has to be solved numerically. Especially the social force model has been investigated, extended, and applied in numerous papers. The basic idea had been stated before by various authors, who, however, then did not translate the social forces to physical forces in pedestrian simulations. In other cases, they *did* use them as physical forces but worked in a different field than pedestrian dynamics. I addressed the conceptual difficulty of directly translating social forces to physical forces. A powerful feature of force-based models is that physical forces, such as contact forces, can easily be introduced. Social forces do not qualify as a plausible decision-making process on the psychological layer.

Alternative models (section 3.4) stem from a variety of fields. In animation and robotics, path planning has been researched for decades. Especially steering behaviours and velocity-obstacle models have found their way into computational science. Agent-based approaches often use existing locomotion models to introduce individual differences among agents in the simulation. However, due to their flexibility, they are often difficult to validate. Some authors have focused on cognitive and social sciences and tried to bring in findings or modelling approaches from these fields. There are some promising approaches that I also refer to later in this work.

Finally, I described similarities and differences, and discussed the concept of scalar fields (section 3.5), which many models have in common. The perspective of scalar fields may be used to help with the choice of an existing model but also to develop new approaches. The limitations found in model flexibility and calibration may have to be overcome with a different concept. Furthermore, the idea of a scalar field that is used as potential, cost, utility, benefit, or probability is not a plausible model for human decision making. This limitation is also addressed further in chapter 6.

Chapter 4

The optimal steps model

The optimal steps model (OSM, Seitz and Köster, 2012) is a microscopic simulation model of pedestrian motion[1]. Agents follow a greedy algorithm to approach a target. Mathematically, it is based on the superposition of scalar fields and a local optimisation scheme. The scalar field is best interpreted as a utility function mapping each point in the plane to a utility value. The local optimisation can then be interpreted as utility optimisation. Although this interpretation is questionable as a representation of human decision-making processes (Gigerenzer et al., 1999), it is accessible to many disciplines, especially in social sciences. In contrast, the concept of potentials and forces may be appealing to physicists but is probably not as accessible to psychologists. The area searched for the optimal next position coincides with the reach of the human step length. Thus, one movement step of an agent in the simulation represents the movement step of a pedestrian. This is intended to bring closer together the physical process of human walking and pedestrian stream simulations.

The dynamic of pedestrian behaviour is simulated in the two-dimensional transverse plane (top-down view). Agents in the simulation are represented as circles with radius b, which is an idealisation of their bodies' extension (Seitz et al., 2015d). Obstacles can have an arbitrary two-dimensional geometrical shape. They do not have any specific height or other attributes. Instead, they are abstract obstacle elements in the scenario that cannot be passed by pedestrians. Targets are geometrical shapes that agents try to reach. Targets are abstract too and may represent a safe place in an emergency scenario or an area pedestrians want to reach in order to remain there.

4.1 Utility functions

Agents in the OSM are guided by a scalar field while approaching the target (Seitz et al., 2016). The scalar field assigns a scalar value to every position in the two-dimensional environment. The values are assessed locally by agents to determine their next motion step. The scalar field is constructed in a way to reproduce emergent effects that match pedestrian behaviour, such as approaching a target and maintaining

[1] Most of the findings I present in this chapter were published as journal articles (Seitz and Köster, 2012, 2014; Seitz et al., 2015b). I refer to the respective papers in the text.

a certain distance to walls and other pedestrians. In psychological terms, this can be seen as an example of approach-avoidance behaviour in the spatial domain:

> Approach motivation may be defined as the energization of behavior by, or the direction of behavior toward, positive stimuli (objects, events, possibilities), whereas avoidance motivation may be defined as the energization of behavior by, or the direction of behavior away from, negative stimuli (objects, events, possibilities). (Elliot, 2006)

The general idea of local optimisation for pedestrian dynamics was first introduced in a simulation with a cellular grid by Gipps and Marksjö (1985). In this simulation, pedestrians can only move from cell to cell, and thus, the scalar values are calculated for the centre of cells, not the whole plane. Gipps and Marksjö (1985) called the various contributions to the scalar field forces that keep agents away from each other. The forces are used to calculate a net benefit for each cell. For every movement step, the agent chooses the adjacent cell with the highest net benefit. The interpretation by the authors as forces ensuring a certain distance between agents is somewhat intuitive, but the interpretation as utility, which is then optimised, seems more accurate. Later, Helbing and Molnár (1995) proposed the social force model, which determines pedestrian motion through equations of physical forces that accelerate the agents. The potentials causing the forces are a scalar field, but in the formulations and implementations of the model, usually only the forces are constructed and not the potentials leading to the forces (Seitz et al., 2016).

In the first publication of the OSM (Seitz and Köster, 2012), we referred to the scalar field as *potential*. This may be misleading as the scalar field is not interpreted as potential causing forces that accelerate the agents like in force-based models. Therefore, in this work, I interpret the scalar field in the OSM as utility function (Seitz et al., 2015b)

$$u : \mathbb{R}^2 \to \mathbb{R}. \tag{4.1}$$

When interpreting the scalar field as utility, repulsive contributions have to be negative and attractive contributions positive. When interpreting the scalar field as potential, it is the other way around.

Three different types of scenario elements contribute to the utility function u: the target, obstacles, and other agents. The overall utility is calculated as the sum of the target utility u_t, the smallest of the m (negative) obstacle utilities $u_{p,j}$, and the sum of all n pedestrian utilities $u_{p,i}$:

$$u = u_t + \min_{j \in 1 \dots m} u_{o,j} + \sum_{i=1}^{n} u_{p,i}. \tag{4.2}$$

The function u can be evaluated for arbitrary points $x \in \mathbb{R}^2$ in the transverse plane. The contributions from obstacles and other agents only depend on their distance d to the focal position x. Hence, u_p is symmetric around the position of the pedestrian (figure 4.1).

In the model we proposed first (Seitz and Köster, 2012), contributions from pedestrians and obstacles to the overall scalar field are based on a negative exponential

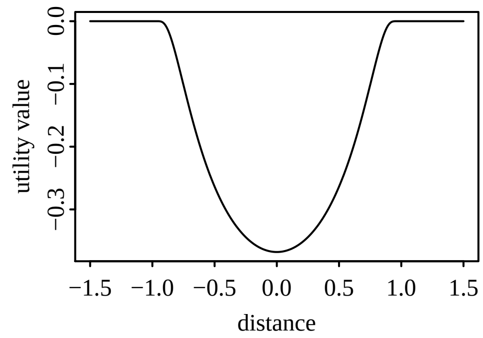

Figure 4.1: Schematic depiction of a utility function around other agents in the OSM. An agent is located at distance = 0 having a repulsive effect on other agents in both directions: the closer the position, the stronger the repulsion gets. This is represented by a stronger negative utility value for closer positions. Beyond a certain cut-off value (1.0 in this case), the agent has no repulsive effect on others, and thus the utility value is 0. In the model, the utility function is two-dimensional and symmetric independently of the orientation. (Figure: Seitz et al., 2015b)

function. This has some disadvantages. First, the function has to be cut off at one point since its support is not bounded and hence would lead to computational difficulties otherwise. Second, due to the cut-off, the function is not continuous. Third, the function has many parameters, which means the model is less parsimonious. Therefore, we employed a function with compact support for both the obstacle and pedestrian utility in a later publication of the OSM (Seitz et al., 2015b), which I discuss in the following.

The distance $d \in \mathbb{R}^+$ between agent positions x_i and x_j is $d = \|x_i - x_j\| - 2b$, where b is the radius of agents, assuming they all have the same size. If agents had different radii b_i and b_j, the distance would be $d = \|x_i - x_j\| - b_i - b_j$. Given the distance, the pedestrian utility is defined using only two parameters h and w (Seitz et al., 2015b):

$$u_p(d) = \begin{cases} -1000 & \text{if } d \leq 0 \\ -h \ \exp\left(\frac{1}{(d/w)^2-1}\right) & \text{if } 0 < d < w \\ 0 & \text{else.} \end{cases} \tag{4.3}$$

If the distance d between agents were negative, they would overlap, and therefore a strongly negative utility is added to prevent this. If $0 < d < w$, a negative utility is added to keep agents away from one another. The parameter w determines the reach of the utility function and h the strength. Outside of the support of the function, the utility is 0 and thus has no effect. Figure 4.1 shows a schematic depiction of the function. The function has various advantages: it has compact support, depends only on two parameters, and is smooth for $d > 0$.

The obstacle utility has the same form as equation (4.3). The distance d is defined as the distance to the closest point of the obstacle to x minus the agent's radius b. The parameters for u_p and u_o must be calibrated independently for obstacle and pedestrian repulsion. The target attraction u_t can be modelled as the negated distance from x to the target if there are no obstacles on the path. This results in increasing utility when agents get closer to the target. When there are obstacles on the path, agents would not skirt them properly but move up to them until the obstacle repulsion sets in. In the worst case, they get trapped by the obstacle.

There are two common solutions to this problem. The first is to place intermediate targets in the scenario so that the path between two targets never crosses an obstacle (Arikan et al., 2001; Kneidl et al., 2012; Kneidl, 2013). This visibility graph can also be generated automatically (Höcker et al., 2010). The alternative we chose for the OSM is a static navigation field (Kretz, 2009; Hartmann, 2010). To obtain the navigation field, the arrival time of a wave front emanating from the target is computed. The negated arrival times can be used as utility function and combined with the other utility contributions from obstacles and agents. I give more detail on the latter approach in subsection 4.1.2.

4.1.1 Parameter calibration

Two different methods can be used for the calibration of repulsive and attractive effects in scalar fields. First, one can employ aggregate measures such as the density-speed relation or evacuation times in certain scenarios to minimise the difference between simulation outcomes and empirical observations (Davidich and Köster, 2012; Davidich and Köster, 2013). Second, one can try to reproduce the individual pedestrian behaviour in the simulation (Johansson et al., 2007; Moussaïd et al., 2009b; Seer et al., 2014). For the first publication of the OSM (Seitz and Köster, 2012), we chose parameters in a negative exponential function that yielded plausible qualitative simulation outcomes. Agents were explicitly decelerated dependent on the local density to match a density-speed relation from literature. Here, I describe another calibration study (Seitz et al., 2015b) where we used both data from individual motion and aggregate data to calibrate the compact support function from above.

For the calibration of w and h in the pedestrian utility function, we used the density-speed relation. We carried out 25 simulation runs of a corridor scenario and measured the density and speed (more details on density and speed measurement methodology can be found in the publications by Seyfried et al., 2005, Jelić et al., 2012a, and Flötteröd and Lämmel, 2015). All parameter combinations for $h \in \{0.1, 0.58, 1.05, 1.53, 2.0\}$ and $w \in \{0.1, 0.33, 0.55, 0.78, 1.0\}$ were used to produce one diagram for each set of parameters. Finally, we chose the one that matched the reference curve reported by Weidmann (1992) best. The result with the selected values $w = 0.33$ and $h = 1.05$ is shown in figure 4.2. Our calibration study was a proof of concept since the density-speed relation varies across scenarios (Chattaraj et al., 2009). There are many different diagrams, and the parameters may have to be recalibrated for every scenario.

For the calibration of w and h in the obstacle utility function, we conducted a controlled experiment with 29 university students. They were instructed to individually walk around a corner of concrete walls in a university building (figure 4.3). We video recorded the scene and later analysed the footage. The resulting trajectories are shown in figure 4.4, and the distance participants kept from the wall is shown in figure 4.5. It can be seen that pedestrians kept the least distance at the corner. Before and after the corner, the distance increases but then seems to converge to a constant value

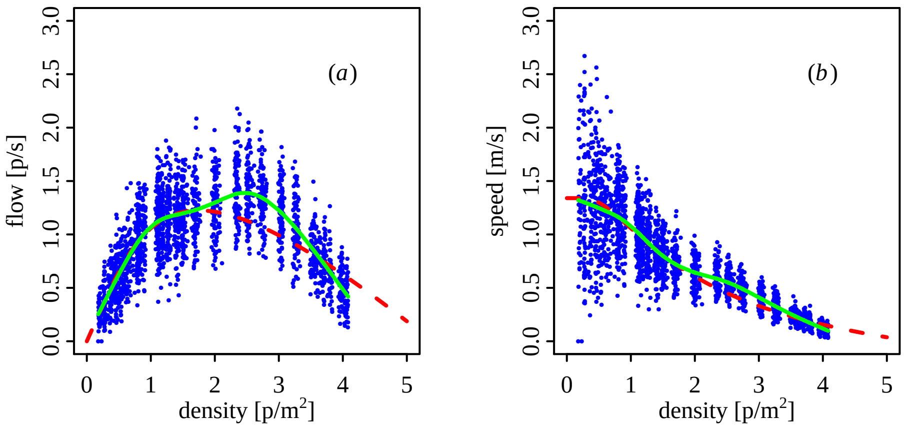

Figure 4.2: Simulated density-flow and density-speed relations for the parameters $w = 0.33$ and $h = 1.05$ in the pedestrian utility function u_p for the OSM. The red dashed line is the reference curve given by Weidmann (1992). The green solid line is a spline regression through the measurement points (blue dots). The measurement was taken in a corridor scenario using methodology described by Steffen and Seyfried (2010) and in our own work (Seitz and Köster, 2014). (Figure: Seitz et al., 2015b)

of about 0.85 m.

To reproduce this behaviour, we chose the parameters for the obstacle utility function accordingly. The Parameter w controls the reach of the obstacle's influence to the agent's edge. Thus, $w + b$ is the reach of the utility when considering the agent's centre. We set $b + w = 0.85$ m with $b = 0.2$ and $w = 0.65$. To account for individual differences, a uniformly distributed error term between -0.2 and 0.2 is added to w for every agent in the simulation. The second parameter $h = 2$ determines the strength of the obstacle repulsion. We chose it in a way that ensures that agents can still pass through narrow corridors but at the same time keep the distance w when possible. In section 4.5, figures 4.37 and 4.38, I report simulation results that were produced using this calibration.

Figure 4.3: Experimental set-up for the parameter calibration of the obstacle utility. Participants were instructed to walk around the corner in both directions. A start and finish line was marked with white masking tape on the floor (here, highlighted in red).

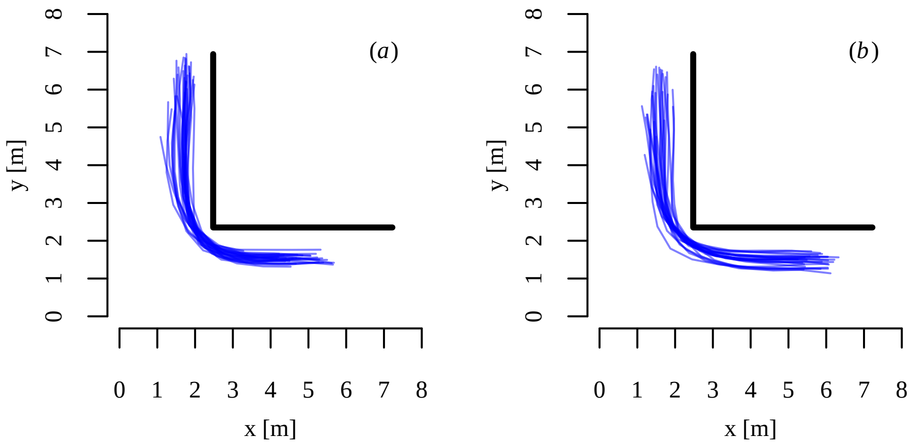

Figure 4.4: Trajectory of pedestrians walking around a 90° corner of concrete walls in a controlled experiment. In experiment a, participants started on the bottom right and in experiment b, on the top left. A start and finish line on the floor was marked with white masking tape. Figure 4.3 gives more details on the experimental set-up. (Figure: Seitz et al., 2015b)

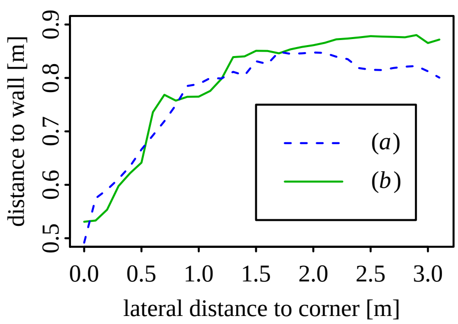

Figure 4.5: Distances pedestrians kept in a controlled experiment when walking around a corner of concrete walls. In experiment a, participants started on the bottom right and in experiment b, on the top left. Participants kept the least distance at the corner. Having passed the corner, the distance participants kept increases but remains constant after about 1 m. Figure 4.4 shows the trajectories of the experiment. (Figure: Seitz et al., 2015b)

4.1.2 Navigation fields

The target attraction can be obtained by measuring the negated direct distance to the target (section 4.1), which is based on the assumption that pedestrians choose the shortest path to the target minimising energy consumption (Kuo, 2001) and time. However, if the direct Euclidean distance is used, agents may get trapped by obstacles (figure 4.6 shows such an obstacle).

As an alternative, one can compute the travel time to the target around obstacles. An efficient method for this is Dijkstra's algorithm (Dijkstra, 1959) on an equidistant grid with grid points as the vertices of the graph. Every vertex is connected through an edge to its neighbours in the von Neumann or Moore neighbourhood. For every grid point, the travel time to the target is computed with Dijkstra's algorithm and stored in an array. Bi-linear interpolation can be used to obtain the target utility at arbitrary positions between grid points. The distance between two grid points is a parameter to the discretisation.

Dijkstra's algorithm does not produce balanced travel times in all directions from the target. Obtaining accurate travel times from every point in the plane to the target can be achieved by computing the arrival time of a wave front emanating from the target. Mathematically, this is expressed by a non-linear partial differential equation – the eikonal equation. It can be solved numerically with the fast marching method developed by Sethian (1996, 1999), which is highly accurate and efficient in computation.

Using such navigation fields for pedestrian motion was first proposed for cellular automata (Kretz, 2009; Hartmann, 2010). With bi-linear interpolation, they can also be used for any other model relying on agent navigation through scalar fields. In the OSM, agents following a navigation field computed with the fast marching algorithm take the shortest path around obstacles (figure 4.6).

The static navigation field does not take into account other agents, and hence, all agents will try to follow the shortest path ignoring congestions, which can lead to unrealistic behaviour when alternative routes are available. The alternative routes may be slightly longer (in distance to walk) but are faster for agents if the shortest way is congested. Agents can be made to choose alternative paths if the shortest route is blocked. For this, the wave front has to propagate slower at densely occupied areas (Kretz, 2009; Hartmann, 2010; Kretz et al., 2011; Hartmann and Hasel, 2014), which also mitigates congestions at corners (figure 4.7). Interestingly, rewarding positions of other agents by making the wave front propagate faster can be used to simulate queueing behaviour (Zönnchen, 2013; Köster and Zönnchen, 2014).

4.2 Local optimisation

Given the utility values at all positions in the transverse plane, agents follow the scalar field to advance towards the target. The target utility ensures that agents finally reach their destination. Obstacle and pedestrian repulsions may divert agents from the shortest path or delay their arrival. Obtaining the fastest path would require global optimisation taking into consideration other agents' motion and utility contri-

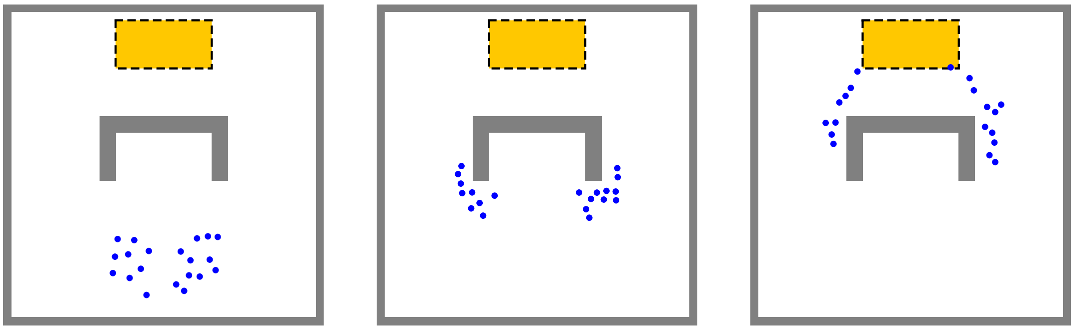

Figure 4.6: Agents following the target utility in the OSM computed with the fast marching algorithm (Sethian, 1996). The side lengths of the room are both 20 m. Agents (depicted as blue circles) walk from the bottom around the obstacle to the (dashed, yellow) target area. Snapshots from left to right show the simulation state after 10, 50, and 100 seconds in simulated time. If the negated direct Euclidean distance to the target were used, agents would get trapped by the obstacle because they ignore that there is no feasible way approaching it directly.

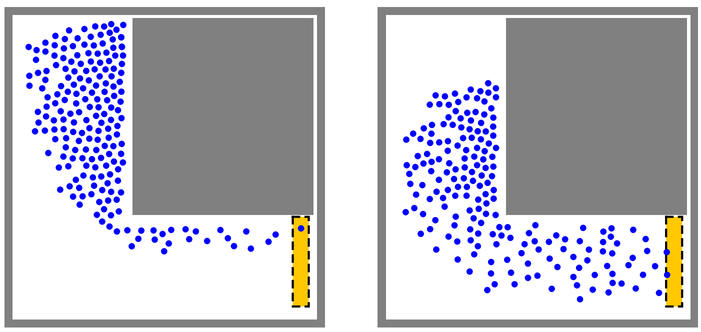

Figure 4.7: Comparison of the static navigation field (left) and dynamic navigation field (right) taking into account other agents on the path (Kretz, 2009; Hartmann, 2010). The side lengths of the room are both 20 m. 200 agents were created on the upper left and approach the (dashed, yellow) target on the bottom right. The snapshots for both figures were taken after 20 s in simulated time. In the left figure, a static navigation field is computed with the fast marching algorithm (Sethian, 1996) ignoring agents in the scenario. All agents choose the shortest path around the corner, and thus, a considerable congestion forms. All agents have reached the target after 88 s. In the right figure, the navigation field is recomputed during the simulation and takes into account other agents. Therefore, travel time is increased for areas that contain other agents. Agents avoid areas that are already occupied by many others and choose wider paths around the corner. All agents have reached the target after 47 s and thus are much faster.

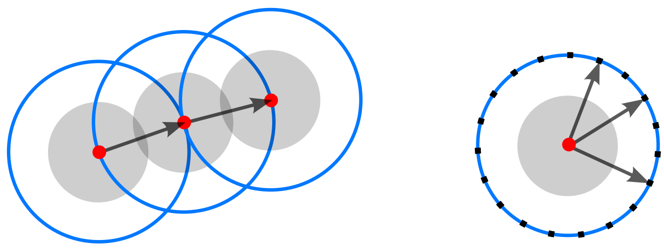

Figure 4.8: On the left, two possible motion steps in the OSM are shown. The red dot represent the position of the agent and the blue circle the step circle to which the agent can move. The arrows illustrate the motion steps. The grey discs represent the agent's body, which cannot be stepped on by other agents. On the right, a grid discretisation as possible optimisation scheme is shown. The agent can only move to grid points on the step circle. (Figure: Seitz and Köster, 2012)

butions. Global optimisation is very demanding in terms of computation time and is implausible in terms of necessary cognitive effort in pedestrians (Gigerenzer et al., 1999). Therefore, agents in the OSM follow a greedy strategy to reach their target.

The next movement step of agents in the simulation is searched for locally at distance r from the current position (Seitz and Köster, 2012). For this, a circle is placed around the current position of the agent and the next position is chosen on the circle (figure 4.8, left). When maximising utility, a one-dimensional optimisation problem with periodic boundary conditions has to be solved.

A robust and computationally efficient numerical solution to this problem can be obtained by placing an equidistant grid on the circle (Seitz and Köster, 2012). Evaluation of the utility at each grid point and selection of the highest utility yields the next position (figure 4.8, right). The number of grid points q is a parameter to the numerical solver and can have a systematic effect on emergent behaviour such as evacuation times. Thus, it has to be chosen with care. To prevent systematic effects because of the placement of the grid, it is rotated randomly. I give more details on the numerical discretisation in subsection 4.2.2.

In cellular automata, there is a fixed number of movement directions. With a grid of squares, there are either four or eight directions: four if movement is allowed in the von Neumann neighbourhood, that is, to cells in vertical or horizontal direction only; and eight if movement is allowed in the Moore neighbourhood, that is, to cells in the von Neumann neighbourhood and to cells in diagonal direction. Six movement directions are available in cellular automata with hexagonal cells.

The optimisation scheme with step circles in the OSM can be adjusted to reproduce the movement behaviour of agents in cellular automata (figure 4.9). The cellular automaton with rectangular cells and von Neumann neighbourhood corresponds to the OSM with one circle and four grid points placed at directions 0, $\pi/2$, π, and $\pi \times 3/2$. The cellular automaton with Moore neighbourhood correspond to the OSM with an additional, larger step circle (larger radius r) and four grid points placed at directions $\pi \times 1/4$, $\pi \times 3/4$, $\pi \times 5/4$, and $\pi \times 7/4$. The cellular automaton with hexagonal cells corresponds to the OSM with one step circle and six grid points placed at directions 0, $\pi \times 1/6$, $\pi \times 2/6$, $\pi \times 3/6$, $\pi \times 4/6$, and $\pi \times 5/6$. To reproduce the

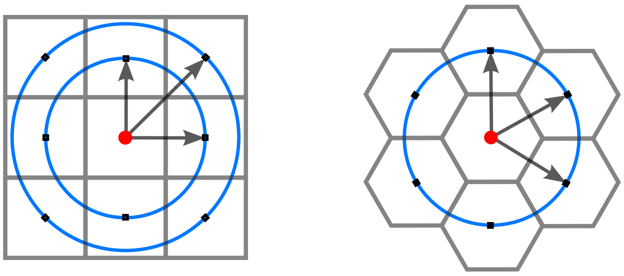

Figure 4.9: Movement directions in cellular automata with corresponding step circles and discretisation in the OSM. Movement behaviour in cellular automata can be reproduced by the OSM if the grid is chosen to match the centre of cells in cellular automata and is not rotated randomly. (Figure: Seitz and Köster, 2012)

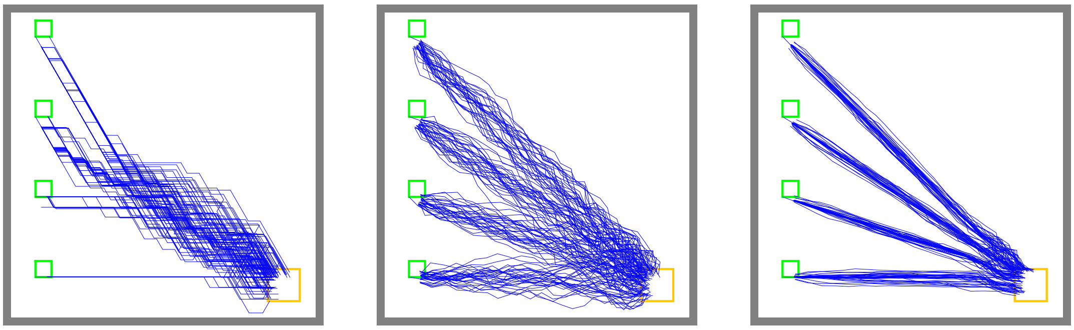

Figure 4.10: Trajectories of simulated agents with different step circle optimisation. Agents are created on the left at four sources. In the left part of the figure, 6 grid points without the random offset are used, which corresponds to the behaviour of hexagonal cellular automata. In the middle, 6 grid points with random offset are searched for the next position. On the right, 50 grid points with random offset are used. The systematic bias due to the fixed directions can be seen when no random offset is used. The same number of grid points with the random offset does not produce any directional bias. With more grid points, the trajectories become more straight and smooth.

movement of the cellular automaton, the grid must not be rotated randomly. Figure 4.10 shows a comparison of emergent trajectories with different grid discretisations.

4.2.1 Step length to speed relation

The natural movement of pedestrians is stepwise. If the step circle's radius r is chosen to match the step length of pedestrians, the motion step length in the simulation coincides with the one of real pedestrians (Seitz and Köster, 2012). As the step length of agents in the OSM is not bound to a grid like in cellular automata, individual agents can have different step lengths. It is well known that the step length depends on the speed of motion (Grieve and Gear, 1966; Kirtley et al., 1985; Fukagawa et al., 1995), which we used for the OSM.

The step lengths of pedestrians can easily be measured in controlled experiments. We conducted an experiment in which participants were instructed to walk, jog, and

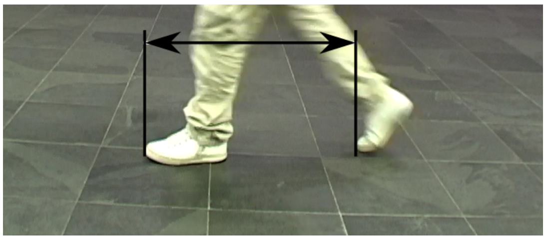

Figure 4.11: Measurement of step lengths in video footage from a controlled experiment. (Figure: Seitz and Köster, 2012)

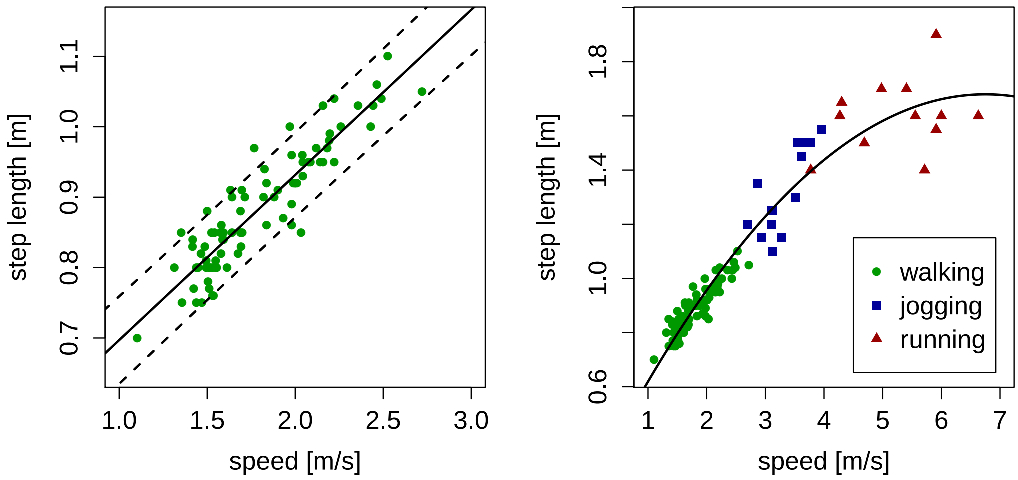

Figure 4.12: Step length to speed relation measured in a controlled experiment. The left part of the figure shows data collected from an experiment where participants were instructed to walk. The observations show a linear relationship. The right figure shows the some data and observations where participants had to jog and run the same distance. Here, the relation seems to suggest a non-linear behaviour for higher speeds. However, there are only a few observations in this regime and the variance is higher, and hence, the regression is less reliable. (Figure: Seitz and Köster, 2012)

run a short distance with self-selected speeds (Seitz and Köster, 2012). The whole experiment was video recorded from the side (figure 4.11). We measured the participants' speeds and step lengths in the video footage and computed a linear regression (figure 4.12). For normal walking speeds, the relation between speed and step length is linear (figure 4.12, left). For faster speeds, the step length seems to converge to a maximum, which indicates that faster speeds are the result of higher step frequencies (figure 4.12, right).

In a second experiment, we used a different measurement methodology (Seitz et al., 2014b). Again, participants were instructed to walk a short distance with self-selected slow and normal speeds. We recorded the experiment from an oblique camera position above the measurement area and annotated the positions of the foot and the time when the foot first touched the ground (figure 4.19 in subsection 4.2.3). This procedure yields discretised trajectories for the left and right foot. The two trajectories can be

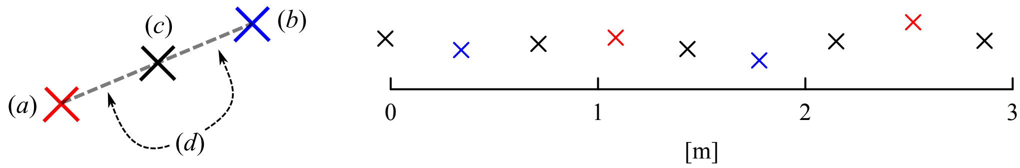

Figure 4.13: Step interpolation for trajectories of the left and right foot annotated in a controlled experiment. The left part illustrates how the centre position (black $\times$) is computed given the left (red $\times$) and right (blue $\times$) foot's position. The right part shows one participant's movement in the controlled experiment with the three trajectories. (Figure: Seitz et al., 2014b)

merged by calculating the centre between the left and right foot after each step (figure 4.13). With this method, the stepwise motion is preserved in the data, both in time and space.

The interpolated trajectory with the centre positions can be analysed to obtain speeds and step lengths. These measurements complement the data from the previous experiment with slower speeds. The results of both experiments are shown together in figure 4.14. The linear relationship still matches the data. The parameters of the regression model deviate slightly from the one with the data from the first experiment alone (shown in figure 4.12, on the left). The regression line for both data sets together has a slope of 0.27 and an intercept of 0.38 (Seitz et al., 2014b). Overall, the second experiment validates the findings from the first one and vice versa.

In the same study, we asked participants to walk sideways, backwards, and suddenly stop during walking. We also reported the resulting speed to step length relations, which could be used in model extensions for specific walking modes. The respective plots are shown in chapter 5, figure 5.3.

The relation between speed and step length can be used in the OSM. At first, agents have to be assigned a preferred walking speed, which they adopt when no other agents and obstacles are present (Seitz and Köster, 2012). The step length is chosen according to the speed as the radius r. Therefore, each agent has a fixed preferred speed and step length. The step length could also be varied according to the current speed, which was not done for the simulations reported in this work.

4.2.2 Numerical discretisation

Finding the next position on the concentric step circle poses a one-dimensional optimisation problem. It can be solved efficiently with an equidistant grid (figure 4.8, right). An alternative is to use a numerical optimisation scheme such as Brent's algorithm (Brent, 1973). Here, I only discuss the equidistant grid because it is very robust and illustrates how the OSM differs from but can also reproduce the motion in cellular automata (section 4.2 and figure 4.9).

Like in cellular automata, a grid with the same orientation would lead to a systematic bias in some movement directions. To avoid the bias, the whole grid can be rotated with a new random offset for every step (Seitz and Köster, 2012). The numerical scheme has only one parameter: the number of grid points q. The direction φ_l to

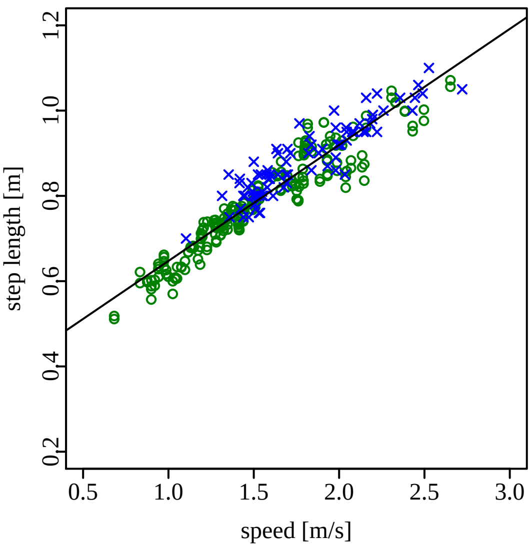

Figure 4.14: Speed to step length relation for normal and slow self-selected speeds of pedestrians in two controlled experiments. The blue crosses show the data obtained first (Seitz and Köster, 2012) and the green circles the data from a second experiment (Seitz et al., 2014b). The regression line has slope 0.27 and intercept 0.38. The linear relation seems to accurately match the data. Nevertheless, a slight decrease of the slope for higher speeds may be present.

the grid point with index l is computed with the formula

$$\varphi_l = \frac{2\pi}{q}(l + \lambda), \tag{4.4}$$

where λ is the uniformly distributed offset with $\lambda \sim U(0,1)$. λ has to be re-drawn for each agent's motion step to avoid systematic numerical errors. Given the current position x_0 and the step length r, the absolute positions x_l of all grid points can be obtained with

$$x_l = x_0 + r \times (\cos(\varphi_l), \sin(\varphi_l)). \tag{4.5}$$

So far, the step length is fixed for each agent. Real pedestrians in dense crowds tend to adjust their step lengths to the current walking speed. To obtain smaller steps, one can search for the optimal position within the step circle (Seitz and Köster, 2012). For example, multiple concentric circles can be used as discretisation together with a grid placed on the circles according to equation 4.4 (von Sivers, 2013; Seitz et al., 2015b). Again, a numerical optimisation scheme such as Nelder-Mead (Nelder and Mead, 1965) may be employed to find the optimum on the disc (von Sivers and Köster, 2015). The two optimisation problems (on the circle and the disc) and the corresponding grid discretisations are illustrated in figure 4.15.

It is important to note that the number q of grid points on the step circle can have a systematic effect on simulation outcomes, including egress times. Therefore, any calibration is only valid for the chosen discretisation with q grid points. Alternatively, a sufficiently large number of grid points can be used to ensure that the emergent effect has converged.

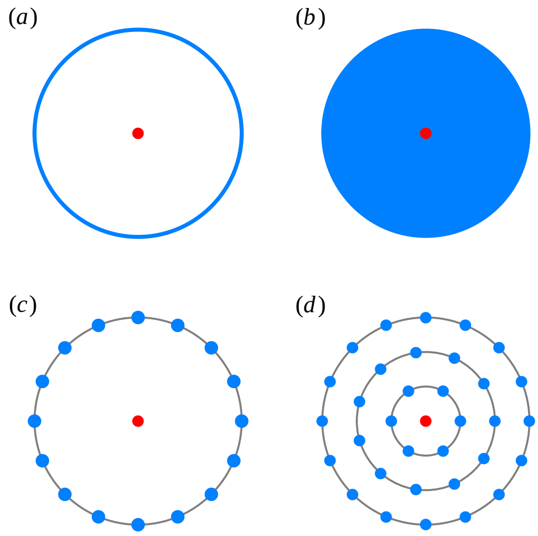

Figure 4.15: Illustration of optimisation schemes for positions on the circle (a) and on the disc (b). On the bottom, corresponding grid discretisations are shown in c and d. (Figure: Seitz et al., 2015b)

I illustrate this with the evacuation time of a room egress scenario similar to the controlled experiment described by Liddle et al. (2009, 2011). In the simulation scenario, 180 agents exit a room through a 2 m wide and 5 m long bottleneck. The time of the last pedestrian who egressed was measured for each run. I conducted the simulation with different numbers $q = 8, 16, 24, \ldots, 300$ of grid points on the step circle. The parameters were set according to the calibration in subsection 4.1.1. I ran the simulation 20 times for each number of grid points q. The result is shown in figure 4.16. From 8 to 50 grid points, the egress time decreases rapidly from about 85 s to about 60 s. From 50 to 200 grid points, the egress time only decreases slightly and does not seem to decrease any further for more than 200 grid points.

Using a different approach for selecting the next position may result in fundamentally different behaviour. Instead of optimising on the circle or disc, it is possible to choose the next position in the direction of the gradient. This is inspired by the gradient navigation model (Dietrich and Köster, 2014) where the gradient is used to steer agents in that direction through determining the velocity. In contrast to the gradient navigation model, the gradient is used to determine the next position on the step circle, which preserves the stepwise motion process and discretisation in time. We referred to this approach as *gradient steps model* in comparison to the gradient navigation and optimal steps model (Seitz et al., 2016).

Since evaluating the utility function is the dominant computational load in the simulation, computational effort goes up with increasing numerical accuracy. Therefore, the complexity of the computation and the computational time necessary to run the simulation strongly depends on the number of function evaluations. Profiling the software with Java VisualVM revealed a strong dependency of the computational time necessary on the number of grid points. I ran the simulation scenario described above one time with 50 and 100 grid points each. In both cases, the utility function

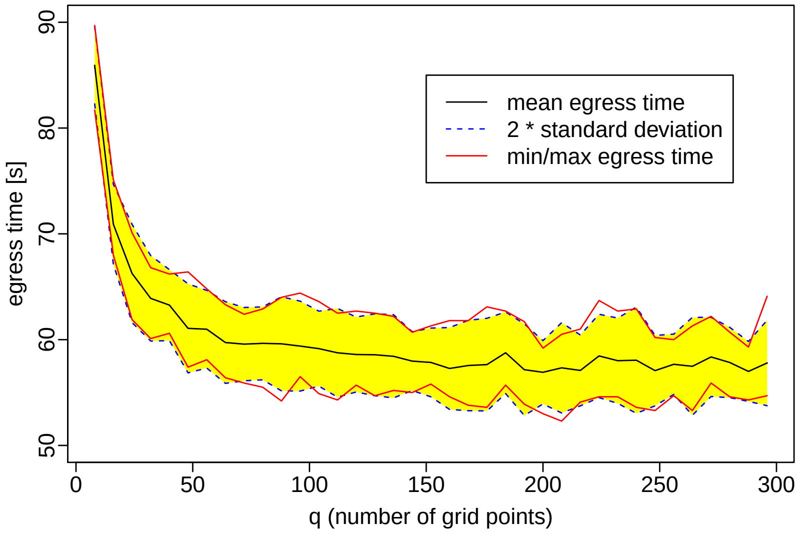

Figure 4.16: Egress times of 180 agents (time it took till the last agent left the scenario) through a 2 m wide and 5 m long bottleneck for different numbers q of grid points on the step circle. The simulation was run 20 times for each $q = 8, 16, 24, \ldots, 300$. The solid (black) line shows the mean egress time, the dashed (blue) lines the mean plus/minus two times the standard deviation, and the solid (red) lines the minimum and maximum egress times of the 20 runs. The figure illustrates how the number of grid points on the step circle can have a systematic influence on emergent behaviour. The egress time seems to have converged at $q = 200$.

evaluation took over 90% of the computation time necessary to simulate the scenario. Therefore, it may be advisable to use fewer grid points for some applications and calibrate the model accordingly when computation time is limited.

4.2.3 Constrained movement direction

The step circle with radius r in the OSM represents the range in which humans may take their next step. Real pedestrians cannot change their direction arbitrarily with any speed. The feasible change of motion direction depends on the current speed of pedestrians. This can be deduced from the fact that a faster change of motion direction requires stronger forces. The force pedestrians can apply is limited and hence how fast they can change motion direction is too. Under normal conditions, pedestrians try to minimise energy consumption (Kuo, 2001); this may additionally limit the speed at which they turn. Therefore, the complete step circle may not be an adequate representation of pedestrian stepping behaviour.

To better match the real pedestrian movement process, the step circle in the OSM can be limited in the direction of the previous motion direction (Seitz et al., 2015b). For slow speeds, the angle of possible directions is greater and for faster speeds it is smaller (figure 4.17). When the pedestrian is not moving, the whole step circle can be used as it is assumed that an arbitrary next direction is possible from this state.

The possible angle for the next step relative to the previous motion direction can be measured in controlled experiments or field observations. For the calibration of

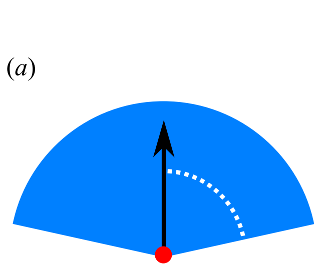

Figure 4.17: Step-constraint approach for the OSM. The red dot depicts the agent's current position and the arrow the previous motion direction and speed. The blue area schematically represents the positions the agent can move to and the dotted white line the maximum angle for the change of direction. With slow speed (a), the agent has a wider possible angle for the next step. For faster speed (b), the angle for the next step is more constrained. (Figure: Seitz et al., 2015b)

Figure 4.18: Experimental set-up for a series of experiments investigating the change of walking direction. Participants had to walk around the obstacles in both directions. In experiment a, the 90^o corner was indicated with white masking tape. In experiment b, the same corner was additionally represented by tables. In experiment c, a straight line was indicated with white masking tape. (Figure: Seitz et al., 2014b)

this parameter in the OSM, we conducted a controlled experiment with 12 participants (Seitz et al., 2014b, 2015b). The participants had to individually walk around obstacles marked with white masking tape on the floor (figure 4.18 shows the experimental set-up). In a control run, they walked a short distance without an obstacle. Then, a 90^o corner was indicated with white masking tape (figure 4.18 a). We used the same geometry again, but this time, additional tables where set up to represent the obstacle (figure 4.18 b). Finally, participants had to walk around a straight line on the floor, performing a 180^o change of direction (figure 4.18 c).

We video captured the experiment and prepared the trajectories as described in subsection 4.2.1 for the second experiment of measuring the step length. An example annotation of the steps is shown in figure 4.19. The resulting central trajectories are shown in figure 4.20. Pedestrians seem to keep slightly less distance to the obstacle before they pass it. When marked with white masking tape, participants cut through the corner, and they kept more distance to the obstacle when it was additionally represented with tables.

The heatmaps in figure 4.21 show areas where the next step relative to the previous direction were taken to with higher frequency. The last step was taken to the origin

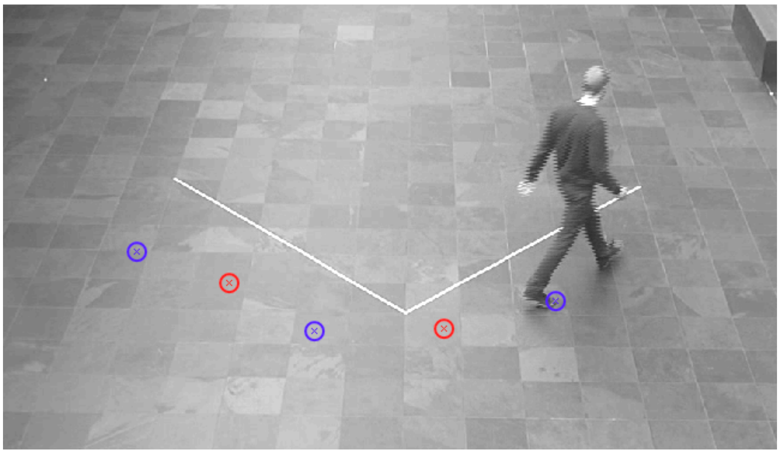

Figure 4.19: Annotation of step positions in the video footage from a controlled experiment. Positions of the left foot are marked with red circles, and positions of the right foot are marked with blue circles.

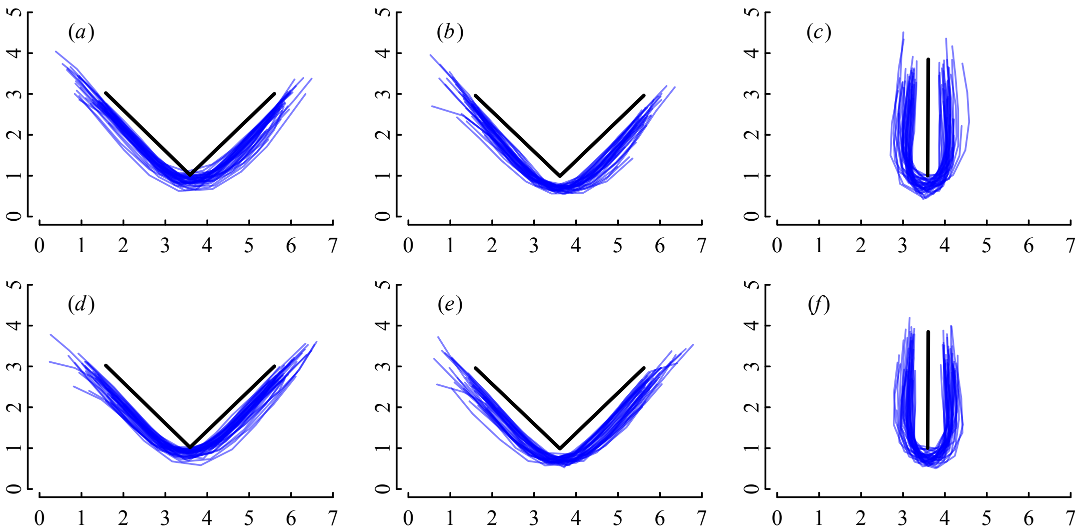

Figure 4.20: Trajectories (in blue) of pedestrians walking around obstacles in a controlled experiment (figure 4.18 shows the experimental set-up). Participants walked from left to right in the upper row, and from right to left in the lower row. The obstacles (in black) were indicated with white masking tape and additional tables in experiments *b* and *e*. Axis values are given in metres. Pedestrians seem to keep slightly less distance to the obstacle before they pass it in experiments *a* and *d*. They also kept more distance to the obstacle additionally represented by tables, while they sometimes actually cut through the corner when the obstacle was only indicated with white masking tape. (Figure: Seitz et al., 2014b)

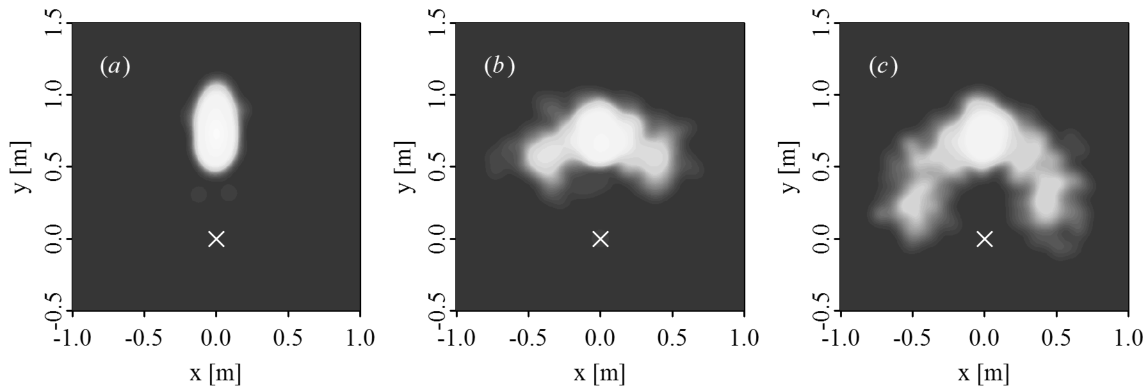

Figure 4.21: Relative positions of the next step to the last step (from the origin, marked with a white ×, in the direction of the y-axis). In a, participants walked a short distance without encountering an obstacle. In b, participants walked around a 90^o corner indicated with white masking tape on the floor. In c, participants walked around a straight line indicated with white masking tape. Figure 4.18 shows the experimental set-up. (Figure: Seitz et al., 2015b)

of the coordinate system in $x = 0$ and $y = 0$ (marked with a white ×) in the direction of the y-axis. The results are shown for the experiment where participants walked a short distance without an obstacle (a), the experiment with a 90^o corner as an obstacle (b), and the experiment where participants had to walk around a straight line (c).

The relation of the angle pedestrians changed the direction with to the current speed for the same experiments is shown in figure 4.22. The plots show how the angle for the change of direction increases the more pedestrians have to turn. A linear regression line indicates whether there is a dependency of the change of direction on the speed. However, since also steps that were taken before and after the corner are included, the crucial effect may not be very pronounced in the regression line. A significant linear relationship was only found for the last experiment (c).

The data of all three experiments is summarised in figure 4.23. For the step constraint in the OSM, I intended to find an upper boundary for the angle given the speed. The angle must never be greater than π and never smaller than 0. Furthermore, when the speed is 0, the angle must be π to meet the requirement previously stated that agents may move in an arbitrary direction if their speed is 0. The red line in the plot is defined by

$$\text{angle} = \pi - \text{speed} \times \frac{\text{s}}{\text{m}} \tag{4.6}$$

and meets all these requirements if the angle is set to 0 for speeds greater than π (with speed measured in metres per second). The right side depends on the unit of the speed. For example, if measured in kilometres per hour, the equation would be

$$\text{angle} = \pi - \frac{\text{speed}}{3.6} \times \frac{\text{h}}{\text{km}}. \tag{4.7}$$

This relation can be used for the constraint of direction in the OSM (figure 4.17). In the simulation, the current speed has to be measured through the movement history of agents. An extension of this model could be to actively slow agents down in order to

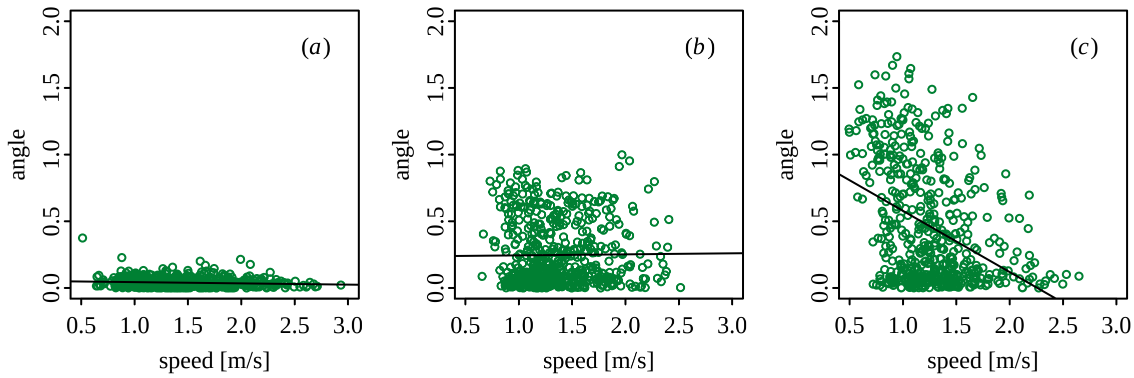

Figure 4.22: Angle of the next movement step relative to the last step in relation to the current speed for three controlled experiments. In *a*, participants walked straight ahead, in *b*, they walked around a 90^o corner, and in *c*, they walked around a straight line, thus had to turn 180^o. The black line shows a linear regression. The dependency of the angle on the speed is statistically significant for experiment *c* and not significant for experiments *a* and *b*, which can also be seen by the slope of the regression line. (Figure: Seitz et al., 2015b)

turn around a pointy corner. This may be a more realistic model of real pedestrians' behaviour, which could also be investigated empirically. In the current model, agents do not slow down first and hence might pass the corner more than necessary because they cannot turn quickly enough given the constraint. However, figure 4.20 suggests that this may actually be the case to a certain degree as pedestrians sometimes seem to keep a greater distance after having passed the corner than before.

The impact of the step direction constraint on egress times in a simple scenario is small. To investigate this, I ran an egress scenario 1000 times for both the model with constrained and without constrained movement. For the optimisation, 50 grid points were used in all runs. I chose the rest of the parameters exactly the same as in subsection 4.2.2 and for figure 4.16. The results are shown in figure 4.24. The mean egress times are 61.44 s and 61.21 s for the simulation runs without constrained and with constrained movement, respectively, and show a statistically significant difference according to a t-test ($p < 0.01$). While the test reveals a significant discrepancy, their means only differ by 0.23 s, which can be considered within a range of indifference and thus to be negligible.

Constraining the movement may still have an impact in other scenarios and on microscopic emergent behaviour. The model could be extended as discussed above to capture the dependency of the speed on the changes in direction. Additionally, the constraint facilitates computation because fewer positions have to be searched for the optimum while sustaining the same numerical accuracy.

4.3 Update schemes

Agents in the simulation have a preferred speed, step length, and next position which is determined by the local optimisation on the step circle or the whole disc. The time

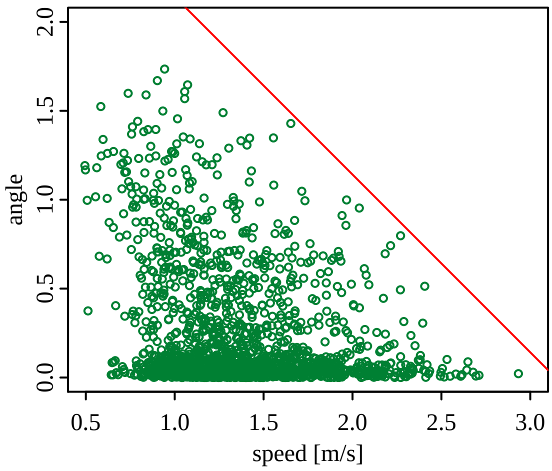

Figure 4.23: All data collected in three experiments (individual data reported in figure 4.22) where participants had to walk around different obstacles, and thus had to turn around to different degrees. The red line shows an upper boundary, which is defined by angle $= \pi -$ speed. This relation can be used in the step constraint model for the OSM (figure 4.17). (Figure: Seitz et al., 2015b)

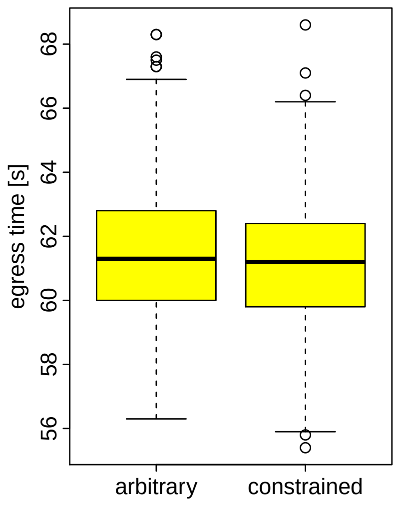

Figure 4.24: Boxplot comparing the simulated egress times for the OSM without movement constraint (arbitrary) and with movement constraint (constrained) of 1000 runs for a scenario with 180 agents. The lower boundaries of the boxes show the lower 0.25 quantile and the upper boundary of the boxes the upper 0.25 quantile. A t-test reveals a significant difference ($p < 0.01$) between the two groups. Nevertheless, the difference between the mean of the two groups is only 0.23 s and thus may be negligible.

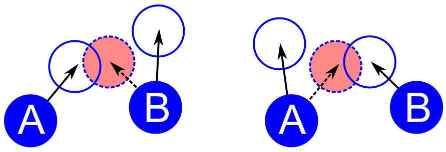

Figure 4.25: Conflict situation where agents A and B both want to move to the centre. The red circles indicate the preferred (counterfactual) position that could not be reached. The solid arrows point to the (factual) next positions. On the left, A moves first and thus B has to evade to the side. On the right, B moves first and thus A has to evade to the side. This illustrates how the order of motion steps in simulated time can have an impact on the individual movement of agents. (Figure: Seitz and Köster, 2014)

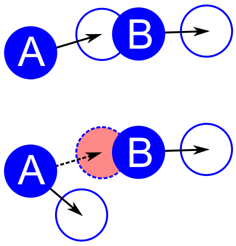

Figure 4.26: Another example of a conflict situation with two agents moving behind each other. The red circles indicate the preferred (counterfactual) position that could not be reached, and the solid arrows point to the (factual) next positions. A evades to the side if B does not move first (at the bottom). If B moves first, A can assume the preferred position behind B (at the top). (Figure: Seitz and Köster, 2014)

it takes to make the next step can be calculated by simply dividing the step length by the speed. What remains is to determine the order in which steps are taken by the agents in the simulation (Seitz and Köster, 2014). I describe various update schemes and discuss differences in the simulation outcome in this section[2].

If two agents A and B who are situated in the same scenario are considered, they may want to move to the same next position. In this case, the agent moving first will occupy the position and the other agent will have to move to an alternative one. This is illustrated in figures 4.25 and 4.26. In figure 4.25, both A and B want to move to the central position. On the left, A moves first, and B evades to the right. On the right, B moves first, and A evades to the left. Figure 4.26 shows another situation where A and B walk to the right, but A has to step to the side if B does not move first.

The examples illustrate that the emergent behaviour of agents in the simulation is influenced by who moves first. Moreover, the collective dynamic of the crowd may change because of the order by which agents take their steps. For example, when agents move in a line behind each other, the motion process is fastest if the individual steps are ordered from front to back.

Robinson (2004) gives a general description of update schemes for computer simulations. There are two basic approaches to process the individual steps (figures 4.27,

[2]The results in this section were published as journal article (Seitz and Köster, 2014).

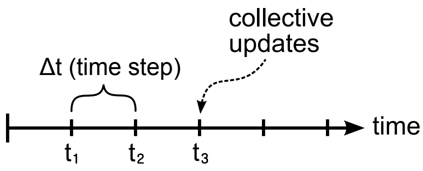

Figure 4.27: Illustration of the time-slicing approach for update schemes. The simulation time t is advanced by a fixed time step length Δt. After each time step, all events that occurred in the last time interval are collectively updated (figure 4.28). (Figure: Seitz and Köster, 2014)

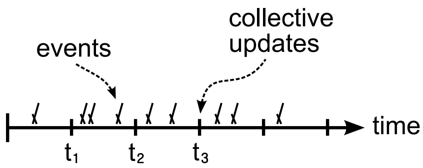

Figure 4.28: Illustration of update schemes with a unit clock and individual events that take place within time intervals. All events in one time interval are collected and updated after the simulation has been advances (figure 4.27). (Figure: Seitz and Köster, 2014)

4.28, and 4.29). The first is the time-slicing approach where the simulation time t is advanced by Δt for all agents. All agents who have finished their previous steps according to their step lengths and speeds within this interval start a new step. The second is the event-driven approach, which is also referred to as discrete-event simulation. Here, agents are moved according to the order of the finish time of their steps. The time-slicing method can also be used for parallel updates. I describe the approaches in the following sections.

4.3.1 Nonparallel unit-clock updates

The time-slicing approach is illustrated in figure 4.27. The simulation time t_0 is advanced by Δt in every time step:

$$\begin{aligned} t_1 &= t_0 + \Delta t, \\ t_2 &= t_1 + \Delta t, \\ &\vdots \\ t_n &= t_{n-1} + \Delta t \quad = \quad t_0 + n \times \Delta t. \end{aligned} \tag{4.8}$$

After n time steps, at t_n, the simulation time has advanced by $n \times \Delta t$ relative to the beginning of the simulation in t_0. All stepping events that have taken place in the last interval are updated collectively after each time step (figure 4.28).

The time δ_i agent i needs to take the step is usually not aligned with the unit clock, that is, does not finish exactly when the time step ends. Steps never end exactly with the time step in the model when preferred speeds are randomly distributed. If $\delta_i < \Delta t$, the agent may move after the first time step from t_0 to t_1 but will also lose $\Delta t - \delta_i$. To retain the preferred speed, the lost time has to be accounted for. One solution is to store an individual time credit τ_i for each agent (Seitz and Köster, 2014). After each time step, the time step length Δt is added to the time credit τ_{n-1} from the last time step with index $n-1$:

$$\tau'_{i,n} = \tau_{i,n-1} + \Delta t. \tag{4.9}$$

If $\tau'_{i,n}$ is greater than or equal to the time it takes agent i to make the step ($\tau'_{i,n} \geq \delta_i$), the motion step is carried out and δ_i is subtracted from the time credit. If the time credit is insufficient for the motion step ($\tau'_{i,n} < \delta_i$), the motion step cannot be made, and the time credit is retained for the next time step:

$$\tau_{i,n} = \begin{cases} \tau'_{i,n} - \delta_i & \text{if } \tau'_{i,n} \geq \delta_i \\ \tau'_{i,n} & \text{else.} \end{cases} \tag{4.10}$$

Given the stepping events after the unit time step has finished, one still has to decide in which order the collection of events is processed. In the original publication of the OSM, we processed events sequentially according to their time of creation, which is a fixed-order sequential update (Seitz and Köster, 2012). This update has the advantage that agents in front tend to not block agents coming from behind. However, it may also be considered rather arbitrary. In contrast to that, the random shuffle update randomly permutes the order in each step (Wölki et al., 2006), which prevents systematic effects due to an order.

Alternative orders for the unit-clock update are worth mentioning. In the frozen shuffle update (Appert-Rolland et al., 2011), the order is randomly permuted once, at the beginning of the simulation and not changed after that. The backward-ordered update (Evans, 1997; Brankov et al., 2004) explicitly orders events spatially: agents at the front move first. This may facilitate the motion process (figure 4.26) but at the same time, it seems questionable whether this update matches natural conflict resolution. The fixed-order sequential update approximates the backward-ordered update if agents at the front are created first. Both the backward-ordered update and the frozen shuffle update are not discussed further in the following.

4.3.2 Event-driven update

The update schemes I described so far each have some theoretical or practical advantage. However, the order does not necessarily match the occurrence of events. The event-driven update processes the motion steps exactly in the order of occurrence. Conway et al. (1959) presented the general idea for computer simulations. Robinson (2004) described the event-driven update, which he referred to as discrete event simulation, for general applications. Bukáček et al. (2014) used an event-driven update, which they called adaptive time span, for their pedestrian simulation model. We introduced the event-driven update to the OSM and as a general update for discrete motion models (Seitz and Köster, 2014). I discuss this approach in the following.

To illustrate the event-driven update, I consider two agents A and B at the beginning of the simulation at $t_0 = 0$ s. Agent A takes $\delta_A = 0.5$ s for a step and B $\delta_b = 0.3$ s. Therefore, A takes one step at $t = 0.5, 1.0, 1.5, \ldots$ and B at $t = 0.3, 0.6, 0.9, \ldots$ in simulated time. Accordingly, the order of agents taking steps is: B,A,B,B,A,... (a similar process is illustrated in figure 4.29). In $t = 1.5$, both agents would take a

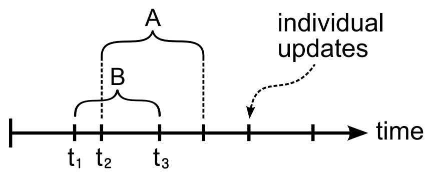

Figure 4.29: Illustration of the event-driven update. Individual motion steps of agents are placed in simulation time. The order is calculated by the time it takes a pedestrian to make the step. No unit clock is required for the processing of events. (Figure: Seitz and Köster, 2014)

step at the same time, and hence, the order is chosen randomly. It is important to choose the order randomly in this case to prevent a systematic effect of the chosen data structure that stores the agents.

The order in which agent steps are processed is their natural order because agents who finish their steps first also have preference in conflict situations. The event-driven update does not need a unit clock nor has the time credit to be stored. An appropriate data structure such as a priority queue (e.g., Cormen et al., 2009, section 6.5) can be used that orders the events. The elements of the priority queue can simply be processed sequentially, and the simulation time is advanced with each event.

It is possible to use the event-driven update together with a unit clock. In this case, the unit clock is advanced as for the unit-clock update (subsection 4.3.1). Events are processed in the same manner as without the unit clock, but the process is stopped after the last event took place within the last time interval. The latter technique may be advantageous if other elements in the simulation require a unit clock or the simulation framework is based on it. In the software framework Vadere, scenario elements such as sources and targets are updated with a unit clock and the simulation loop is also based on it. Therefore, I implemented the event-driven scheme within the unit-clock update.

Another interesting feature of the event-driven update is that other schemes that use a unit clock with collective updates converge to the order of the event-driven update with $\Delta t \to 0$ (Seitz and Köster, 2014). This happens because time steps become smaller and smaller until every event has its own time-step slot. Then, the unit clock imposes the order of the events the way they occur in simulated time. This behaviour can also be observed in the simulation study discussed below in subsection 4.3.4.

Additional computational effort and memory are necessary for the ordering of events in the priority-queue data structure. The calculation of the time of the next event is negligible and a similar computation has to be conducted for the time credit in the unit-clock update schemes. Apart from that, there is no additional computational demand for the event-driven update. Therefore, the event-driven update is computationally almost as efficient as the fixed-order sequential update.

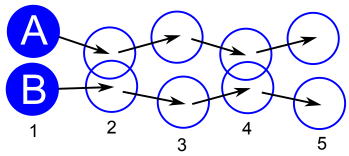

Figure 4.30: Possible motion process with a parallel update. Agents A and B both try to walk in the middle. They do not maintain the minimum distance necessary to avoid collisions because of the parallel update. In step 2, they both move to the middle. In the next step, step 3, they move away from the middle because of the strong repulsion of overlapping physical representations. This can happen multiple times and lead to oscillatory motion. (Figure: Seitz and Köster, 2014)

4.3.3 Parallel unit-clock update

In our publication on update schemes (Seitz and Köster, 2014), we argued that the event-driven update reflects natural conflict resolution best and that it is the preferable choice for the OSM. However, sometimes computational effort can exceed the computational capacities or fast computation of many agents may be necessary. Then, parallelisation of the computational tasks can be a solution. Parallelisation is a problem with all of the update schemes discussed so far because they rely on the information of the steps already taken by other agents. This information is unavailable with parallel updates: agents who are allowed to move in the same time step cannot use the information of other agents' next position. The lack of information can lead to collisions, which can lead to oscillatory motion (figure 4.30).

To prevent collisions, we proposed another mechanism for the OSM (Seitz and Köster, 2014). First, the next positions of all agents allowed to move after a time step are computed in parallel. At this point, agents do not have the information of the next position of one another. Second, all collisions that result from this last update step are identified. Third, only one of the agents in a particular collision remains at the new position, the others are repositioned to their previous location. Finally, the time credit is only reduced by δ_i for those agents who have actually moved to a new position. This algorithm prevents collisions: other agents do not step on any of the previous positions since they use the last state of the simulation for the determination of their next step. The parallel update converges to the event-driven update with $\Delta t \to 0$.

4.3.4 Impact on simulation outcomes

Two simulation studies demonstrate the impact the time step length Δt and the chosen update scheme can have on simulation outcomes. In the first study, 80 agents egress from a room through a bottleneck, and the egress times are measured. In the second study, a corridor is filled with an increasing number of agents, and the density-speed relation is computed. Details on both simulation scenarios can be found in our publication on update schemes (Seitz and Köster, 2014).

I ran the evacuation scenario 500 times for the following update schemes: fixed

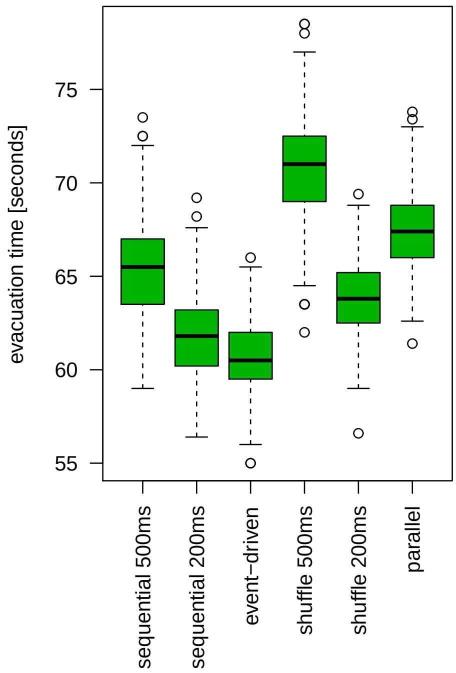

Figure 4.31: Boxplot reporting the egress times in the simulation through a bottleneck with 80 agents for different update schemes. The update schemes yield significantly different evacuation times but they converge to the result of the event-driven scheme for smaller time step lengths Δt. (Figure: Seitz and Köster, 2014)

order sequential update with $\Delta t = 0.5$ s, fixed order sequential update with $\Delta t = 0.2$ s, the event-driven update, the random shuffle update with $\Delta t = 0.5$ s, the random shuffle update with $\Delta t = 0.2$ s, and the parallel update presented in subsection 4.3.3 with $\Delta t = 0.2$ s. The respective egress times are shown in figure 4.31. The egress times are significantly different (t-test, $p < 0.0001$) between all groups. The random shuffle update with $\Delta t = 0.5$ s leads to the longest evacuation times. With the event-driven update, agents leave the scenario fastest. In the group of schemes with $\Delta t = 0.2$ s, the parallel update leads to the longest evacuation times. Both the fixed order sequential update and the random shuffle update approach the evacuation times of the event-driven scheme for smaller Δt, which was expected theoretically.

I ran the corridor scenario with different numbers of agents in the scenario to cover a wide range of densities. The density was measured with Voronoi diagrams (Seitz and Köster, 2014). The same simulation runs were computed for the following update schemes: fixed order sequential update with $\Delta t = 0.5$ s, fixed order sequential update with $\Delta t = 0.2$ s, the event-driven update, and the parallel update presented in subsection 4.3.3 with $\Delta t = 0.2$ s. Figure 4.32 shows the results. On the left, the density-speed relation for the event-driven update is compared to a reference curve reported by Weidmann (1992). On the right, the regression curves for the different update schemes are shown. Again, the event-driven update leads to the fastest speed. Only for low densities, the fixed order sequential update seems to yield faster speeds,

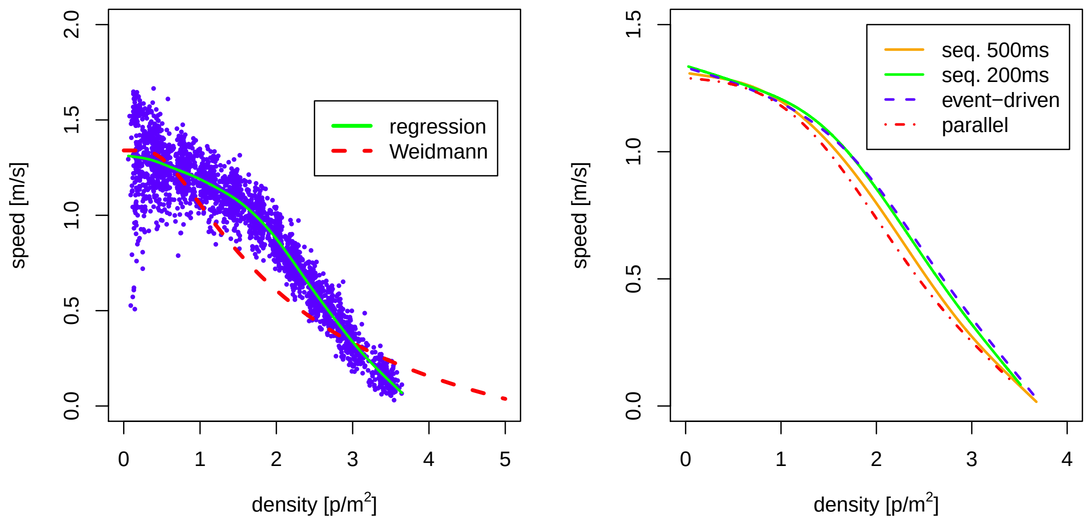

Figure 4.32: Density-speed diagram measured in a corridor simulation scenario for different update schemes. On the left, a scatterplot shows individual measurements for the event-driven scheme and a regression line in comparison to the reference curve reported by Weidmann (1992). On the right, the regression lines for four different update schemes are shown. The event-driven update led to the fastest speeds, although at low densities, the fixed order sequential update seems to yield slightly faster speeds. The parallel update led to the slowest speeds of agents. The shape of the curves is almost identical in all cases. (Figure: Seitz and Köster, 2014)

which may because of random effects. The parallel updated leads to the slowest speeds. The shape of the relations is almost identical across the different schemes.

I demonstrated that different update schemes can lead to different emergent behaviour both on the individual scale as well as in collective measures, including egress times and the density-speed relation. Hence, it has to be noted that any calibration carried out for a model with one update scheme may become obsolete when another update scheme is employed.

4.4 Implementation details

In this section, I describe the software structure used for the implementation of the OSM in the framework Vadere. The simulation loop triggers the object `OptimalStepsModel` to update the agents after each time step by calling the method `update`. The object of type `OptimalStepsModel` holds a list with all agents of the type `PedestrianOSM`, which is a derivative of the general pedestrian class. `PedestrianOSM` has member variables and methods that determine the behaviour of agents. Namely, the method `updateNextPosition` determines the position the agent steps to, and the method `makeStep` carries out the movement. For the determination of the next step, the agent has to know where other agents and obstacles are. Therefore, they have access to the topography object. These relations are shown in the class diagram in figure 4.33.

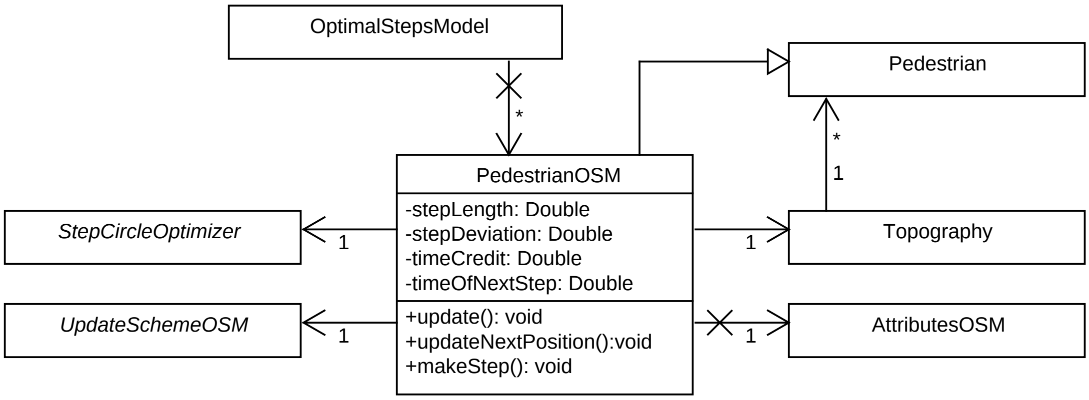

Figure 4.33: Class diagram of the optimal steps model implementation. The simulation loop triggers the update of all agents by calling the function `update` of the object `OptimalStepsModel`. `PedestrianOSM` has access to the positions of other agents through the topography object. Model parameters are set in the class `AttributesOSM`. A variety of optimisation and update schemes can be set through the interfaces `StepCircleOptimizer` and `UpdateSchemeOSM`, respectively.

Different optimisation schemes (section 4.2) and update schemes (section 4.3) can be implemented by using the dedicated interfaces `StepCircleOptimizer` and `UpdateSchemeOSM`. The interfaces allow for the use of a variety of models without interference of one another. The update scheme still has to be known in the class `OptimalStepsModel` because for the event-driven update, an additional priority queue ordering the stepping events is necessary. Figure 4.34 shows implementations of the interfaces that represent schemes I discuss in this work.

4.5 Simulation results

In this section, I report simulation results that validate the OSM as model for pedestrian dynamics. Figures 4.35 and 4.36 show simulation results with the utility functions and calibration we presented in the first publication of the OSM (Seitz and Köster, 2012). The trajectories shown in figures 4.37 and 4.38 were generated using the utility functions and calibration that I described in section 4.1 (Seitz et al., 2015b). In addition, Seer (2015) conducted verification and validation tests for the OSM in another simulation framework and compared the results to other simulation models.

Figure 4.35 shows the trajectories of agents egressing from a room (from bottom to top) through a 5 m long corridor of different widths (0.7 m, 1.0 m, and 2.0 m). The trajectories demonstrate that a varying number of lanes are formed depending on the width of the corridor. A similar behaviour has been observed with real pedestrians (Schadschneider and Seyfried, 2011). When the corridor is very narrow (0.7 m), pedestrians only form one lane in the middle. With a wider corridor (1.0 m), two lanes on both sides of the corridor can be observed, and when the corridor is fairly wide (2.0 m), a third, more fuzzy lane forms in the middle.

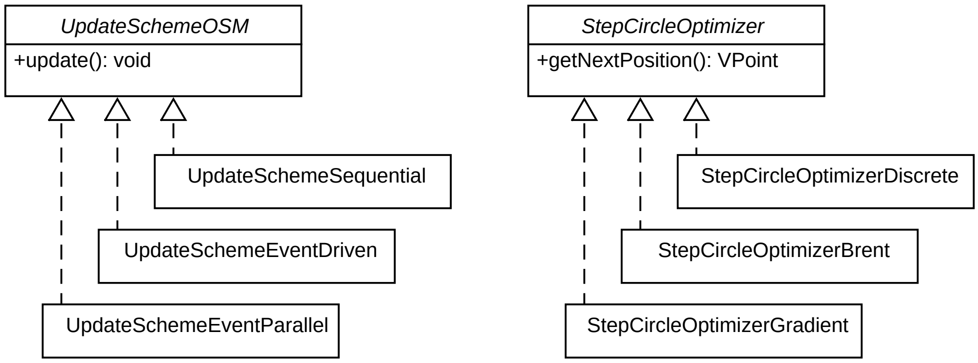

Figure 4.34: Class diagram with implementations of the interfaces `StepCircle-Optimizer` and `UpdateSchemeOSM`. The interfaces allow for the use and implementation of different schemes without changing other classes of the model.

For figure 4.36, a scenario with 180 agents egressing through a bottleneck with two different width (2.0 m and 1.0 m) was simulated. The colour maps show a local measure of occupancy that takes into account agents and obstacles (Seitz and Köster, 2012). Warmer colours indicate higher densities. The panels on the left side show the same situations with agents represented as black circles. The snapshots where taken (from top to bottom) at $t = 10, 20, 30, 40$ s for the left side with a corridor width of 2.0 m and $t = 10, 20, 50, 70$ s for the right side with a corridor width of 1.0 m. This methodology could be used to investigate the design of buildings and discover possible hazards due to high crowd densities.

In figure 4.37, the calibration discussed in section 4.1 was used to simulate the controlled experiment where participants had to walk around a corner of concrete walls. The trajectories observed in the experiment are shown on the left and the trajectories of agents in the simulation on the right. One major difference between the two groups is that real pedestrians tend to get closer to the corner while simulated agents tend to sustain the same distance throughout. This could be a limitation of the simple utility function on complex support. Nevertheless, the average and variance of distances are reproduced well in the simulation.

In figure 4.38, another egress scenario was simulated with different numbers of agents (10, 30, 100). The aim of this study was to demonstrate that agents keep a certain distance to walls but are also able to give up their preferred distance in order to navigate through a corridor. The latter was expected to be more likely with higher densities. In experiment a, 10 agents leave the room from bottom to top through the corridor (with a width of 1.2 m). Only one lane forms in the middle as agents try to keep 0.85 m distance to both sides. In experiment b, 30 agents mainly stay in the middle but sometimes accept positions closer to the wall. In experiment c, 100 agents form two lanes on both sides of the corridor because of the high density.

The studies demonstrate that the optimal steps model can reproduce phenomena that have been observed in real pedestrian crowds. The validation so far is rather qualitative and relies on effects that can be identified in the visualisation of the sim-

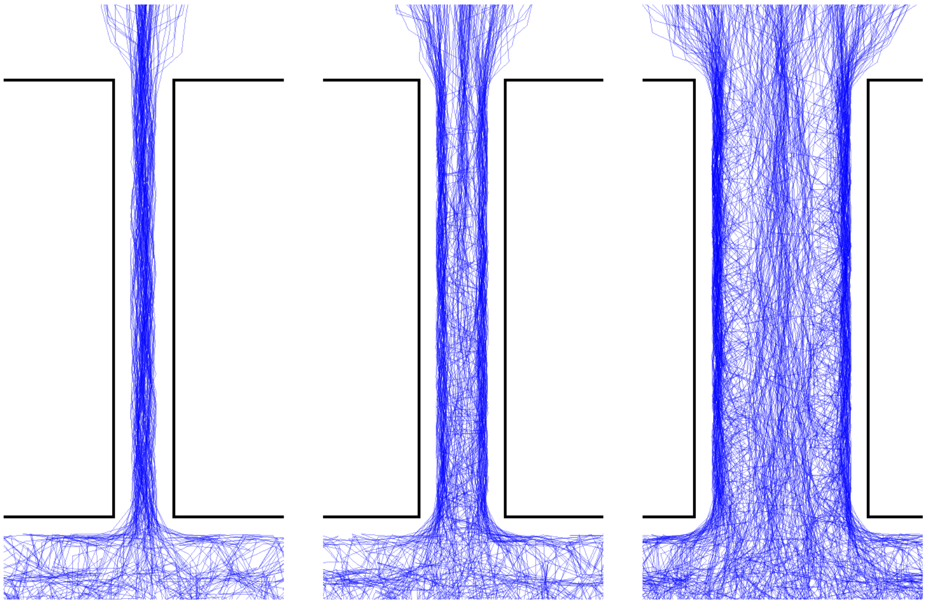

Figure 4.35: Agents egressing from bottom to top through a 5 m wide corridor. The black lines represent the walls, and the blue lines show agents' trajectories. The corridor has widths of 0.7 m, 1.0 m, and 2.0 m (from left to right). A varying number of lanes form for the different widths, an effect that has been observed in controlled experiments with real pedestrians (Schadschneider and Seyfried, 2011). (Figure: Seitz and Köster, 2012)

ulation output. Quantitative measures in pedestrian dynamics are mainly used for the density-speed relation, which can vary greatly in different scenarios. Therefore, it is important to show that different density-speed relations can be reproduced (Seitz et al., 2015b and section 4.1.1). Nevertheless, visual comparison and qualitative phenomena are still crucial validation steps.

4.6 Further developments and utilisation

The optimal steps model has been used for a series of studies, and several further developments based on it have been proposed. In their work on social distances, von Sivers and Köster (2015) studied numerical algorithms for the optimisation of the next step. They optimised for positions within the step circle (section 4.2). They also used a more complex function to represent the repulsion between pedestrians that is based on the concept of social space according to Hall (1966). Köster et al. (2015) presented a model extension for agents walking on stairs. The assumption in this model is that the locomotion is strongly influenced by the individual steps on the stairs (Seitz et al., 2014b). Therefore, the optimal steps model is apt to represent the stepping process on stairs. They also validated the model by comparing it to data collected in a field observation. In his Bachelor's thesis, Zönnchen (2013) implemented a dynamical navigation field and used the optimal steps model for the locomotion of agents (subsection 4.1.2). Going beyond the approach of penalising

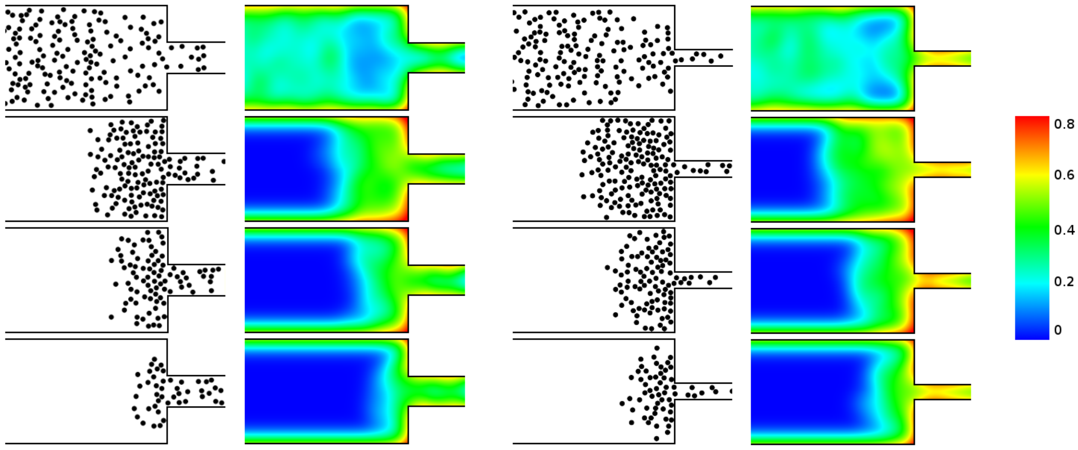

Figure 4.36: Simulation study of 180 agents egressing through a corridor (from left to right) with two different width: 2 m on the left (first and second column) and 1 m on the right (third and fourth column). The first and third column show the state of the simulation at t = 10, 20, 30, 40 s (left) and t = 10, 20, 50, 70 s (right), where pedestrians are depicted as black circles. The colour maps in the second and fourth column show the respective local measures of occupancy taking into account nearby agents and walls. Warmer colours indicate a higher level of occupancy. More details on the methodology used to compute the measure can be found in our publication (Seitz and Köster, 2012). The results show high levels of occupancy and thus density in the corners of the room and in the narrower corridor. (Figure: Seitz and Köster, 2012)

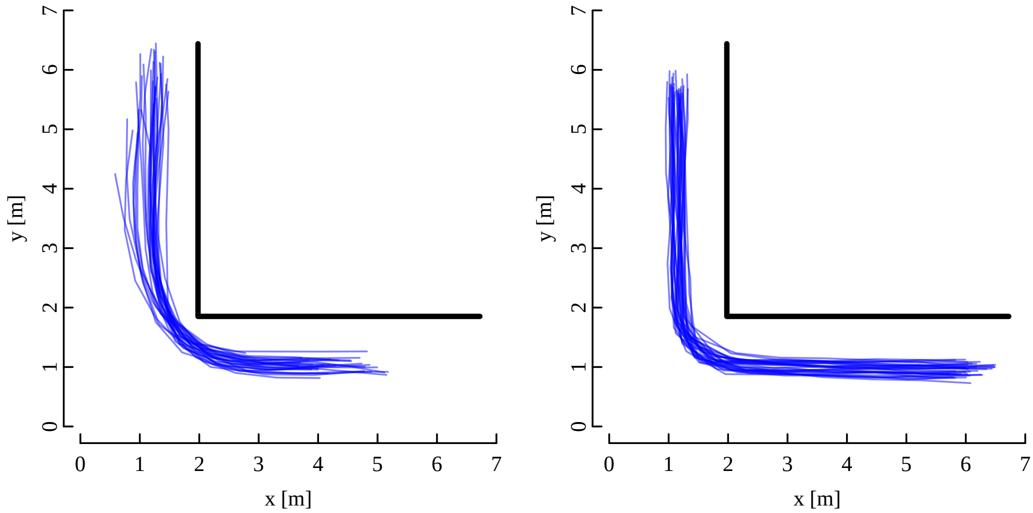

Figure 4.37: Comparison of trajectories observed in a controlled experiment (left) with simulated trajectories (right). For the simulation, the calibration described in subsection 4.1.1 was used. Mean and variance of distances to the wall were reproduced by the simulation. Participants tended to get closer to the walls at the corner in the experiment, which was not reproduced by the simulation model. (Figure: Seitz et al., 2015b)

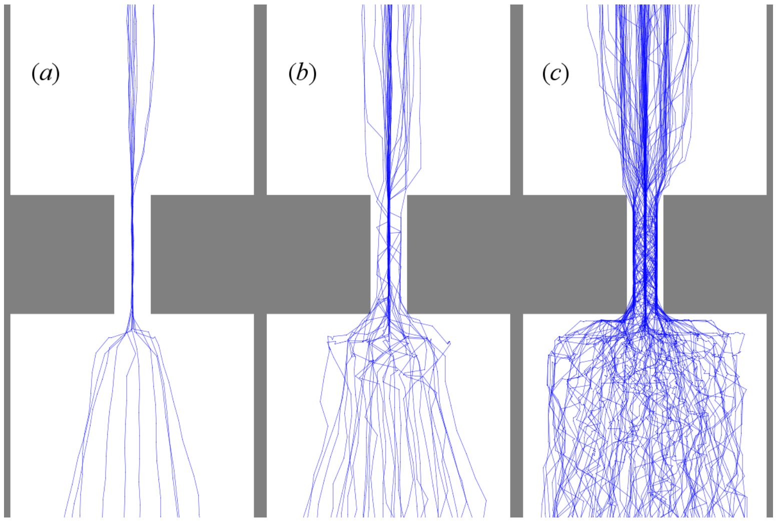

Figure 4.38: Trajectories of a simulated egress scenario. Agents pass from bottom to top through a 1.2 m wide corridor. The number of agents increases from left to right: 10 in part *a*, 30 in part *b*, and 100 in part *c*. With increasing numbers, agents are more likely to accept positions closer to the walls. When there are only a few agents in the scenario, such as in *a*, they can maintain the preferred distance to both sides of the corridor. (Figure: Seitz et al., 2015b)

areas that are occupied, agents can be rewarded for following the path of others (Zönnchen, 2013; Köster and Zönnchen, 2014). This allows for the simulation of queueing behaviours with a variety of shapes. Using this model, behaviours with varying degree of competitiveness can be simulated in the same scenario (Köster and Zönnchen, 2015).

As has been noted many years ago (Sime, 1995), crowd simulation may lack a representation of models from psychology. Following up on this criticism, von Sivers et al. (2014) used the social identity model to develop behavioural descriptions and simulated the egress after the bombing of the London underground in 2005. The scenario was studied empirically, which provided insights into the behaviour of the victims. The authors late quantified the impact of different behaviours. They showed that helping others affects egress times but does not change it much after a certain threshold of the percentage of helping agents is reached (von Sivers et al., 2016). To systematically study the impact of a change in the parameters, they used uncertainty quantification (Smith, 2014), which allows for the efficient estimation of the distribution of an outcome given the distribution of input parameters.

All of the developments mentioned so far were realised in the software framework Vadere after it had been designed and implemented. The use of Vadere for new model aspects and extension demonstrate that the framework is suitable for the use in a research context. In particular, the developments support the argument that it is extendible, modular, and flexible. Seer (2015) implemented the optimal steps model in the extensive simulation framework for pedestrian dynamics used and developed at the Austrian Institute of Technology and ran a series of benchmark scenarios with it. He compared the simulation outcome to other simulation models in his doctoral thesis (Seer, 2015).

The consulting company accu:rate (Kneidl and Sesser, 2015) use the optimal steps model to study public transportation systems and events. They developed their own simulation software that also includes graph-based routeing in interplay with navigation fields. They have extended the model to allow for the simulation of multiple-storey built environments. They also developed extensions for the movement on stairs following Köster et al. (2015), small social sub-groups following our work (section 7.2 in chapter 7), and queueing in front of a service counter. The simulation studies they have carried out include the evacuation of a large-scale event in a harbour, pavilions at a fair, and a lecture hall. Furthermore, the company supported the capacity planning of a museum and assisted in the planning phase of the reconstruction of an underground railway station (Kneidl and Sesser, 2015, description of projects).

4.7 Summary

The optimal steps model (OSM) is a microscopic pedestrian stream model in the transverse plane. It is discrete in both time and space, like cellular automata. In contrast to cellular automata, agents do not move from cell to cell nor is movement bound to a spatial grid of any kind but rather takes place in continuous space. The spatial discretisation represents the natural human step, and hence, it can be considered a natural discretisation. Mathematically, a circle with radius r is placed around agents. This radius has the same size as the step length of real pedestrians and can be determined through controlled experiments (subsection 4.2.1).

To determine where the next step has to be placed on the circle, a scalar field which is being interpreted as a utility function is evaluated (section 4.1). The scalar field is a combination of various contributions from obstacles, other agents, and the target. The target attracts agents, which can be realised with a navigation field (subsection 4.1.2). Navigation fields are scalar fields that have increasing utility when getting closer to the target. The fast marching algorithm allows computing the navigation field efficiently on an equidistant grid. To obtain values for arbitrary positions, bilinear interpolation is used. Obstacles and other agents contribute with repulsive functions. We have published two different functions for the obstacle and pedestrian repulsion in the OSM, the second of which I described in detail.

The utility function I used for pedestrian and obstacle repulsion has compact support and is smooth within a certain range. It is suitable for more advanced optimisation schemes that rely on the existence of the derivatives of the target function. The functions have only two parameters, which makes the model more parsimonious compared to the one in the original publication of the OSM. The parsimony in parameters facilitates calibration (subsection 4.1.1). We calibrated the parameters for the obstacle utility to match the distance pedestrians keep to a wall while walking around a corner. For this, we conducted controlled experiments and analysed them statistically. We calibrated the pedestrian utility through a systematic simulation study by measuring the density-speed relation. A reference relation was selected, and the parameters best matching it were chosen. Different parameter sets should be considered for other scenarios because the density-speed relation strongly depends on the context.

Since finding the position for the next step can be formalised as an optimisation problem, different numerical solutions are possible. I mainly used an equidistant grid on the step circle (subsection 4.2.2). The grid has to be rotated randomly in order to prevent systematic effects known from cellular automata. If of interest, the movement behaviour of cellular automata can be emulated with a fixed grid (without random offset) and a specific number of grid points (section 4.2). This shows how the OSM can be considered a generalisation of cellular automata. The only advantage cellular automata seem to have is that they can be more efficient in computation. The cellular grid is an implicit spatial data structure that always provides the relevant neighbours in constant time. However, this is not the dominant computational load in the OSM. Utility-function evaluations are the major computational load, and, therefore, the OSM can be considered almost as efficient computationally as cellular automata.

With a sensitivity analysis, I demonstrated the impact the number of grid cells has on emergent effects (subsection 4.2.2). Egress times through a corridor decreased with increasing number of grid cells but seemed to have converged at around 200 grid points. This has to be considered when calibrating the model. After calibration, the number of grid points should not be altered arbitrarily as this may introduce a systematic bias. Therefore, either a sufficiently great number of grid points must be used or the number of grid points has to be kept constant after calibration.

Real pedestrians cannot change their direction of motion arbitrarily. They either have to slow down or change the direction slowly. We investigated this in a series of controlled experiments where participants had to walk around obstacles that made them turn around to different degrees. A descriptive statistical analysis revealed an upper threshold for the angle of the next step relative to the last step, which depends on the speed. I used this relation in the OSM to constrain the step circle in a way to prevent angles greater than those observed empirically (subsection 4.2.3). Although the constraint had little effect on egress times in a simulation study, it is a first step in introducing biomechanical findings into pedestrian simulation models.

The events of agent steps have to be processed either in parallel or sequentially. In the latter case, conflicts can be prevented, but events have to be scheduled somehow (section 4.3). I discussed various update schemes (subsections 4.3.1 to 4.3.3) that also showed a systematic effect on egress times in simulation studies (subsection 4.3.4). The event-driven approach (subsection 4.3.2) processes events in the order of their occurrence, which can be considered their natural order. Sequential schemes and random shuffle updates converge to the event-driven result for decreasing time step lengths $\Delta t \to 0$. Sometimes, real-time or faster-than-real-time requirements may demand parallel processing of events. I presented an update scheme that reverses collisions that occurred because of a loss of information in parallel updates (subsection 4.3.3).

In section 4.4, I described how the OSM was implemented in the software framework Vadere. An agent representation allows for collecting the routines necessary for computing individual movement steps in the class `PedestrianOSM`. Two interfaces facilitate the generic use of a variety of update and optimisation schemes without changing the basic implementation of the model. I discussed simulation results (section 4.5) that reproduce important phenomena observed in real pedestrian behaviour, which validates the model.

Finally, I discussed studies and models that are based on the OSM in section 4.6. A series of extension that has been realised in the framework Vadere illustrates the extendibility, modularity, and flexibility of the software design. The extensions also show that the basic model is apt for a variety of contexts going beyond the original application. Furthermore, the simulation model can be used for the study of built environments to improve their safety, efficiency, ease of use, and comfort, which is demonstrated by the consulting firm accu:rate, who use the OSM in their simulator.

Part II

Towards a natural physical and psychological process

Chapter 5

The physical layer: Pedestrian locomotion

In the first part of this work, I discussed approaches to microscopic pedestrian stream simulation based on principles that describe crowd phenomena. The models included important ideas and components that are necessary for any pedestrian simulation, but limitations were also identified. In this part, I present a simulation approach with three layers that focuses on capturing the underlying processes of pedestrian dynamics. The first layer – covered in this chapter – is dedicated to the physical representation and motion of pedestrians (figure 5.1). The second layer (chapter 6) describes individual decision processes and the third layer (chapter 7) group behaviour.

The second and third layer are interconnected because individual and group behaviour cannot be separated clearly, which I also discuss in chapter 7. The physical and psychological layer are interconnected in the real world but can be better separated in a model representation. As pedestrians, we usually have a good idea of what will happen when we take a step in a particular direction. Thus, we have intuitive knowledge about the physical world which, in combination with perceptual cues, affects our decisions. With our movement decisions, we actively change the physical world, at least, the positions of our own bodies.

The interaction of the mental being with the physical world is a research field

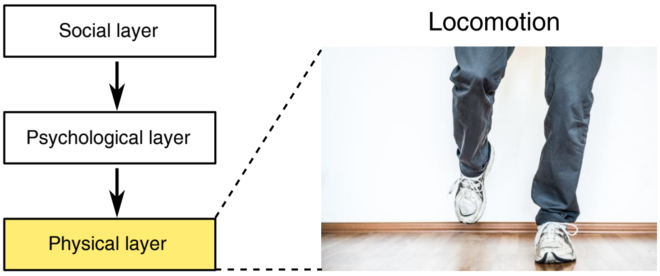

Figure 5.1: Illustration of the model separation into three layers. This chapter focuses on physical aspects and provides a basis for the psychological and social layer.

of its own (Clark, 1997). For example, the mind can be seen as extending beyond the body using the environment (Clark and Chalmers, 1998). More specifically, in embodied cognition (Wilson, 2002; Anderson, 2003; Adenzato and Garbarini, 2006) the functioning of cognition in the physical world is investigated. It shows how the mind is strongly related to the physical world and that a separation in layers is an idealisation of reality. Nevertheless, for the purpose of this work, it is useful and hence justified as a model. The separation allows for the dedicated investigation of the locomotion process with known models from physics. Having a separate layer facilitates the introduction of findings from biomechanics and allows for rather independent modelling of the decision-making process. Computational solutions and software components can be better modularised, studied, and exchanged.

The purpose of the physical layer is to represent the physical process of pedestrian dynamics. Pedestrians move in the environment with a continuous motion process, which is determined by physical forces. The forces stem from their interaction with the ground through friction and from gravity. Contact forces may be present when pedestrians touch an object in the environment or collide with another pedestrian. In very dense crowds, physical contact can entail phenomena like crowd turbulences (Helbing et al., 2007). Looking at the motion process in more detail, pedestrians move by stepping forward, which also constraints the freedom of motion. Ultimately, the physics of human movement codetermines the motion of pedestrian crowds.

A variety of domains can be studied to gain insights into the physical movement of pedestrians. Physics describes the motion of objects and how they are affected by forces. The physical basis for biological systems is studied in biophysics (e.g., Cotterill, 2002). The mechanics of biological movement processes is studied in biomechanics (section 5.1), also by using computational methods (e.g., Anderson et al., 2007). In robotics (section 5.2), biomechanics is of interest when engineering a bipedal robot (e.g., Buschmann et al., 2015) because one has to understand how walking works before it can be replicated in a machine.

Physics, biomechanics, and robotics may provide findings that are useful for a specific simulation approach in pedestrian dynamics. All of them are extensive fields and, therefore, only a fraction of them can be covered in this work. I discuss the aspects that seemed most important to the topic of simulation models in pedestrian dynamics. However, I cannot claim completeness in reviewing neither of the fields, and only some aspects can be considered in the simulation models proposed in section 5.3. The main reason for this is that the complexity of the models must be kept low to allow for meaningful verification and validation. It is also a general objective to choose simple models over complex ones if available (section 1.3 in the introduction). Finally, models should be simple in terms of computational demand to allow for better investigation of the models and for large-scale predictions.

The physical layer is dedicated to the locomotion process of pedestrians. The focus lies on aspects that may affect pedestrian dynamics or aspects that are of interest for a specific scenario. The idea of a separate physical representation of locomotion is not new. In game development, the use of a physics engine is common, and the physical layer can be considered a physics engine. For example, Reynolds (1999) described such an approach for the animation of characters.

I already introduced the stepwise motion process of humans for the optimal steps model, which I discussed in chapter 4. Discrete stepwise motion is one approach to the physical representation of the human locomotion process. The second approach is based on forces. When using a force-based model for locomotion, the position, speed, and acceleration is calculated. The forces acting on the body and the mass of the body have to be known to calculate the acceleration. Solving the second-order differential equation describing the acceleration yields the speed and position of the bodies. Phenomena like motion and inertia can be observed if the system is simulated over time.

Once a basic force-based physical layer is realised, models from biomechanics and other force-based models can be introduced. A force-based model allows for detailed studies that require a physical model or some aspects of it, including crowd turbulences or the impact of pedestrian behaviour on the built environment (e.g., Bocian et al., 2012; Carroll et al., 2013). In general, such a simulation can be extended for many applications. It gives researchers in biomechanics and similar fields a tool to test their models in the context of pedestrian dynamics.

5.1 Aspects from biomechanics

In this section, I discuss aspects of biomechanics for the simulation of pedestrian dynamics. Biomechanics can be defined as "the study of the structure and function of biological systems by means of the methods of mechanics" (Hatze, 1974). In pedestrian dynamics, only the biomechanics of humans is of interest. Winter (2009), who wrote a textbook on the biomechanics of human movement, writes, "The biomechanics of human movement can be defined as the interdiscipline that describes, analyzes, and assesses the human movement" (Winter, 2009, p. 1).

The focus of this section is on walking. Walking can be considered controlled falling (e.g., Lacquaniti et al., 1999) because while walking, we position the centre of mass in a way that we move forward mainly because of gravity. "Since the limbs (or segments) are changing their positions while a pedestrian is walking, the center of mass of the human body does as well" (Winter, 2009, p. 88). This is illustrated by the idea of passive dynamic walking:

> The human frame is built for walking. It has both the right kinematics and the right dynamics—so much so, in fact, that our legs are capable of walking without any motor control. Their gait can be sustained simply by interaction of gravity and inertia, in a natural limit cycle which we call *passive dynamic walking.* (McGeer, 1993)

Bipedalism (moving on two feet) is not exclusive to humans. For example, it can be observed in other primates and birds. It is still not entirely clear how bipedalism evolved in humans (Hunt, 1994, Richmond et al., 2001, and Schmitt, 2003 investigated the evolution of bipedalism in humans). A more recent theory states that bipedalism in primates may also have evolved for the purpose of moving in trees (Thorpe et al., 2007). In any case, walking largely defines our natural locomotion today and thus codetermines pedestrian dynamics.

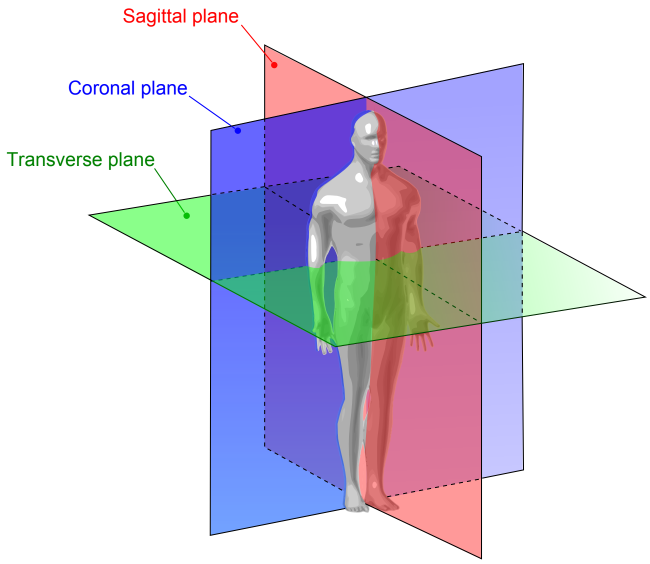

Figure 5.2: Body planes (or anatomical planes) commonly used to describe human movement (e.g., Knudson, 2007, chapter 3). The simulation of pedestrian dynamics is usually two-dimensional and situated in the transverse plane. The two other planes, however, are more common in simulation models of biomechanics. (Figure: Wikimedia, Bouza, 2012)

Biomechanics can be important in simulation models of pedestrians. However, simulating the whole body with all its features is intractable for pedestrian dynamics with many individuals. Important aspects of biomechanics have to be selected. For this work, I propose two models for the physical layer: a discrete stepping process and a continuous force-based process. The former captures the stepping behaviour of real pedestrians and the latter the continuous motion and abstract forces acting on the body. The remaining aspects of biomechanics discussed in this section may serve as background information or basis for future developments.

Physiological properties of the human body are studied in detail in anthropometry, which is a discipline of biomechanics (Winter, 2009, chapter 4, pp. 82–106). The height and weight give the most basic description of a person's physiology. For a force-based locomotion model, the mass is especially important because it is necessary to compute the acceleration given the forces acting on the body. For the simulation of pedestrians, other measures can be relevant, such as the physical fitness or any disabilities affecting the walking capabilities.

Many other properties may have an influence on pedestrian behaviour. Additional physiological properties include age, look, and smell. Dynamic features could be exhaustion, alcohol level, or drug influence. If such features are considered, it is important to properly define how they actually affect pedestrian behaviour. For example, when considering age, one could assume an older person is slower than a younger person. In that case, it may be better to directly measure the speed over the considered population rather than the age. If only the age distribution in a population is known, regression could be used to deduce the speed distribution; but again, the simulation model would only use the speed distribution.

According to the scope of my work (section 1.4 in the introduction), only two-dimensional simulation models for pedestrian dynamics are considered. Real human locomotion and biomechanics are situated in three-dimensional space. Figure 5.2 shows a common definition of body planes (or anatomical planes) that divides the three-dimensional space into two-dimensional planes (e.g., Knudson, 2007, chapter 3). The top-down view is referred to as the *transverse plane*, the view from the side as the *sagittal plane*, and the view from the front as the *coronal plane* (or *frontal plane*). Infinitely many two-dimensional planes could be placed in the three-dimensional space, but these are the ones commonly used in biomechanics. Furthermore, they correspond to a coordinate system that describes arbitrary points in three-dimensional space. The transverse plane is usually used for simulations in pedestrian dynamics.

There are various areas in biomechanics that can be of interest when studying pedestrian dynamics (e.g., Winter, 2009). Kinematics is the study of movement, that is, the position, velocity, and acceleration. It does not, however, consider forces. The internal and external forces are studied in kinetics:

> Internal forces come from muscle activity, ligaments, or the friction in the muscles and joints. External forces come from the ground or from external loads, from active bodies (e.g., those forces exerted by a tackler in football), or from passive sources (e.g., wind resistance). (Winter, 2009, p. 10)

There are two different classes of simulation models (e.g., Winter, 2009). The first are inverse simulation models, which try to reconstruct the forces that led to a particular movement. Forward models, on the other hand, predict movement based on the knowledge of the forces. Movement is predicted based on a model of the body in the synthesis. Given such a model, one can predict the physical behaviour of individual pedestrians. For the purpose of pedestrian dynamics, forward models and the synthesis of human movement are of particular interest.

A variety of simulations for human locomotion has been presented in biomechanics that are usually situated in the coronal or sagittal plane. Winter (2009, p. 202–203) gives a brief review of forward solution models. One of the early works simulating bipedal walking of primates dates back to 1979 (Yamazaki et al., 1979). Probably the simplest model is the inverted pendulum theory (Cavagna and Margaria, 1966), which can be related to dynamic walking (Kuo, 2007). More complex approaches include segment models (Koopman et al., 1995) and detailed muscle models (Martin and Schmiedeler, 2014), which are computationally costly. In contrast to pedestrian dynamics, the mechanics of individuals are the focus and not interactions or collective behaviour. Nevertheless, a detailed model of individual movement may contribute to the study of pedestrian dynamics.

The optimal steps model already has some features of biomechanics: it uses the step lengths of humans for the discrete positions in the transverse plane. The relation between speed and step length is considered (subsection 4.2.2, chapter 4), which we measured in a controlled experiment (Seitz and Köster, 2012). Furthermore, we conducted experiments to investigate the change of direction depending on the speed and used the findings in the optimal steps model (Seitz et al., 2014b, 2015b and subsection 4.2.3, chapter 4). These empirical and model studies all concern the kinematics of pedestrian motion in the transverse plane.

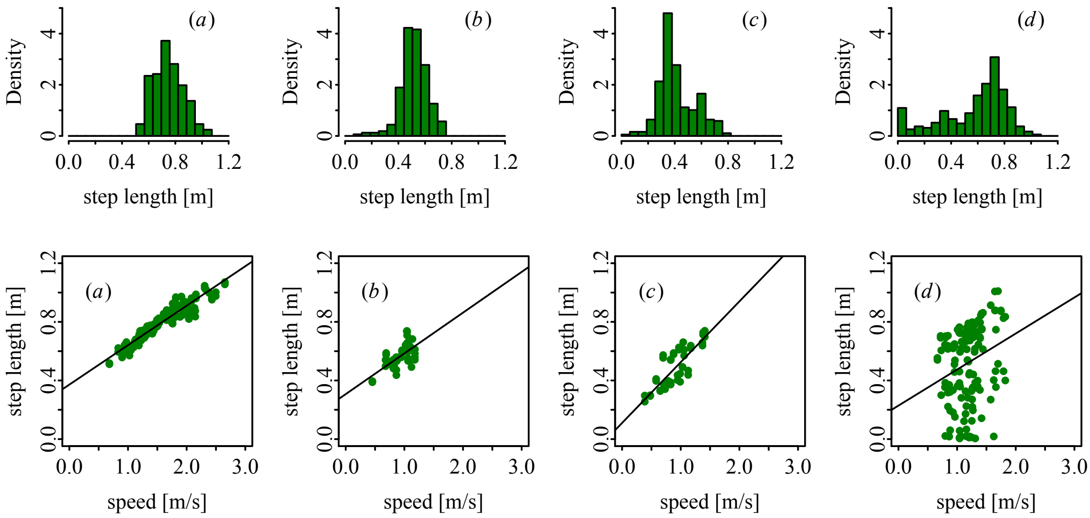

Figure 5.3: Histogram of step lengths (top row) and speed to step length relation (bottom row) in four controlled experiments. In experiment a, participants had to walk forwards, sideways in b, backwards in c, and walk forwards and then suddenly stop in d. Very short steps in d are observed because of the sudden stopping. Apart from d, step lengths are in a certain range and the regression does not vary greatly. Although the variation in speed may be too low for stable regression lines, the observation indicates a more general relation between speed and step length across different modes of locomotion. (Figure: Seitz et al., 2014b)

In another series of controlled experiments, we studied the relation between speed and step length depending on the current mode of locomotion (Seitz et al., 2014b). Participants had to walk forwards (a), backwards (b), sideways (c), and walk forwards and then suddenly stop (d). The step length and the relation to speed are shown in figure 5.3. Interestingly, except in experiment d, the step length stays in a certain range independently of the mode of locomotion and the regression line does not vary greatly. Although the variation in speed may not be sufficient to yield reliable regression lines, the findings indicate that there is a more general optimal relation of speed to step length.

In the same paper, we analysed the stepping behaviour on stairs while participants walked up and down in a series of controlled experiments. We found that participants stayed in the middle of the stairs and mostly took one stair at a time, even when instructed to move fast. Köster et al. (2015) measured stepping behaviour again and developed an extension for the optimal steps model that reproduces it.

Many models of human movement in biomechanics seem too detailed or complex for the simulation of pedestrian crowds. Most biomechanical models are situated in the sagittal or coronal plane, while in pedestrian dynamics, mostly the transverse plane is of interest. Although some aspects of human movement are well-understood, others are not. There is still no complete model that reliably predicts human walking movement across scenarios. Nevertheless, there is a wealth of research that can be drawn on and some models – also in the sagittal and coronal plane – may be relevant for specific scenarios of pedestrian dynamics.

The optimal steps model allows introducing findings from kinematics about stepping behaviour into models of pedestrian dynamics. The decision-making layer could be complemented with a more detailed kinematic model of stepping. This may be especially useful for studying the impact of stepping behaviour on the built environment, such as the excitation of bridges (Macdonald, 2009). The force-based model presented in section 5.3 allows for the introduction of kinetic models and could be combined with a more elaborate representation of walking mechanics.

5.2 Aspects from robotics

Studies in biomechanics mostly aim at a better understanding of the mechanical processes. They do not necessarily target predicting or simulating movement. In robotics, the principle goal is to build a machine that moves in a specific way. Therefore, a strong focus is put on engineering. In order to build (engineer) a walking robot, one has to understand its mechanics and thus understand walking (e.g., Buschmann et al., 2015). In this section, I discuss robotics as a possible source of findings that can be used for the simulation of pedestrian behaviour on the locomotion layer.

Humanoid robots are a special class of robots that are designed to imitate human behaviour. The research objective is to match natural behaviour or aspects of it. Findings in this area can be of interest for pedestrian simulation, especially advances in bipedal walking, collision avoidance mechanisms, and route choice behaviour. I discuss the decision-making aspects of these topics in section 6.2 of the next chapter. An example of a bipedal robot that can walk forwards and backwards, run, and climb stairs is ASIMO (figure 5.4). It has been developed by Honda Motor Co., Ltd. (2004) since 1986 and was first presented in 2000. Another humanoid robot, Atlas, is developed by Boston Dynamics (2016). The most recent version of this bipedal robot can walk in rough terrain and interact with the environment. For example, Atlas can open doors, lift and carry a box, recover balance after having been pushed, and stand up after having fallen to the ground.

Robots are based on a similar concept as situated agents in artificial intelligence. They perceive the environment through sensors, make decisions according to some rules, and the robot's mechanic carries out actions. I cover both perception and decision making in chapter 6. How to control the legs or other mechanical entities of the robot (e.g., Pratt et al., 2001; Westervelt et al., 2007; Hamon et al., 2014) may be useful knowledge for the simulation of pedestrian locomotion. For example, the concept of dynamic walking (McGeer, 1993) is interesting as a biomechanical model for the locomotion layer of individual agents. The approach has already been used to build robots – both passive ones that do not need any motor control and active ones that can walk forward on even ground (Collins et al., 2005; Iida et al., 2007; Rushdi et al., 2014; Zhu et al., 2014).

Dynamic walking robots seem to capture the natural gait of humans especially well. They built on the fundamental mechanics of walking and are validated through the construction of physical models (robots) that actually walk. As I argued is the case for most models in biomechanics, models of dynamic walking are too complex at present for the representation on the physical layer of pedestrian dynamics. Nevertheless, findings from robotics could become valuable in future developments.

Figure 5.4: An example of a bipedal walking robot. ASIMO can walk, run, walk backwards and climb stairs. (Figure: Wikimedia, Vanillase, 2011)

5.3 Locomotion models for pedestrians

I describe two approaches for a physical layer of pedestrian locomotion in this section. The first one stems from the optimal steps model and is the basis for simulating pedestrian dynamics with cognitive heuristics (subsection 6.3.1 of the next chapter). This concept does not consider forces but represents the natural stepping process of pedestrians. The second one is a force-based model that simulates continuous motion in continuous space without considering steps. It can be combined with both the decision making of the optimal steps model and cognitive heuristic. The two alternatives demonstrate how the psychological (decision-making) layer is relatively independent of the physical (locomotion) layer.

5.3.1 Discrete stepping process

The optimal steps model (chapter 4) is based on a discrete representation of pedestrian stepping behaviour, which is a discrete physical representation of natural pedestrian kinematics. The movement is continuous in space and hence does not show any grid artefacts. Cellular automata do not represent the stepwise movement of pedestrians but only a rather arbitrary discretisation, and the cell size is not chosen to match the human step length but their physical dimensions. I first briefly revisit the stepping process and discretisation approach and then point out how it can be used by a psychological layer.

In the discrete stepping process, individual pedestrian motion is represented by the stepping behaviour of humans. The position of the next step is determined by a decision-making process and then the pedestrian is directly moved there. The process

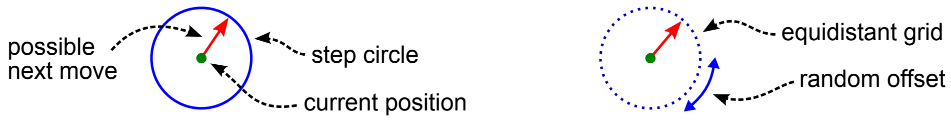

Figure 5.5: Schematic depiction of the step circle in the optimal steps model. The green point represents the current position, the blue circle positions the agents can step to, and the red arrow one possible movement. The continuous model is shown on the left. Steps are taken instantly at one moment in simulation time. The step length, that is, the radius of the step circle is determined through the speed. On the right, one numerical discretisation is depicted with an equidistant grid on the step circle. The grid has to be rotated randomly in order to prevent a systematic bias in the step direction. Details on the optimal steps model can be found in chapter 4. (Figure: Seitz and Köster, 2014)

is discrete because the path from one position to the next is not simulated. The agent is moved there directly and instantly at one point in simulation time. However, it is *in* continuous space and time as steps can be taken in arbitrary directions in the transverse plane and at arbitrary points in the simulated time.

The stepwise movement process is exploited for decision making. Decision-making by agents is only necessary when a step has to be taken by an agent. This is a simplification of the real behaviour of pedestrians because a decision might also be taken between two steps. Nevertheless, it seems likely that the position of the next step is already decided on when the person starts carrying out the step, and the decision only changes in special situations while the step is being made. One example of an exception could be the sudden stopping we studied in a controlled experiment, which I reported in section 5.1.

With the event-driven update scheme, conflicts do not occur or, in other words, are solved implicitly. More details on update schemes can be found in chapter 4, section 4.3. The stepping process can be parallelised, although this entails some modification to the conflict resolution behaviour of agents. The event-driven update scheme represents a natural discrete model. If two agents want to go to the same position, the agent who starts moving there first is allowed to take it, and the other agents must choose another position.

In the optimal steps model, the next step is determined through utility optimisation on a circle or disc (figure 5.5). Instead of utility optimisation, other methods to choose the next step can be used. For example, the gradient steps model (section 3.5) determines the direction of the next step through the gradient of the scalar field at the current position. We used the described representation of the stepping process for cognitive heuristics on the psychological layer (Seitz et al., 2015a): heuristic decisions are made for every new step. This is efficient in terms of necessary cognitive effort, which is one of the requirements for a plausible representation of human decision making (section 6.1.3), and it is efficient in terms of computational effort.

Due to the separation of layers, findings related to the stepping behaviour used in the optimal steps model can also be used in combination with cognitive heuristics. For example, it is well-known that the step length of pedestrians is correlated to their speed (Grieve and Gear, 1966; Kirtley et al., 1985; Fukagawa et al., 1995; Seitz and

Köster, 2012). Other examples are constraints in the direction of movement (Bauer and Kitazawa, 2010; Seitz and Köster, 2012), movement on stairs (Köster et al., 2015), smaller steps (von Sivers and Köster, 2015), and adjusted step lengths for other modes of locomotion such as walking sideways and backwards (Seitz et al., 2014b). Finally, the step length can be related to other measures, including density (Jelić et al., 2012b), which opens up a broad field of research.

The stepping process allows for the introduction of features from biomechanics, yet it is a mere kinematic model with discrete steps. The stepping process can be made continuous in several ways, for example, the steps can be interpolated. A force-based model may be used to realise the step to the next position that has been chosen by a decision-making layer. A more concise approach to a force-based locomotion layer does not explicitly represent the steps but still uses the decision-making layer that assumes a stepwise movement. Such a concept is proposed in the following subsection.

5.3.2 Continuous force-based process

In this subsection, I propose a force-based model for the locomotion layer. In contrast to the force-based models discussed in chapter 3, my approach does not model the decision making with forces. Instead, all forces in this model represent physical forces, which are caused by gravity, friction, and physical contact. The model I describe in the following does not capture these forces in detail nor does it implement contact forces between pedestrians. It does, however, provide a basis for a more detailed representation of the physical world with continuous space and time. It is also independent of the decision-making layer as it allows for the combination with different models, in particular with cognitive heuristics (chapter 6). Thus, it is a proof of concept for genericity of the physical layer as well as a continuous alternative to the discrete process of the last subsection.

A requirement for the model is that the decision *where to go* be already made and only carried out by the physical layer. Moussaïd et al. (2011) also separated the decision-making layer from the physical layer and used a force-based model for locomotion. The basic formula of mass m times the acceleration $\ddot{x}$ equals the force f is used:

$$m \times \ddot{x} = f \;\Rightarrow\; \ddot{x} = \frac{f}{m}. \tag{5.1}$$

Given a model of the force and the mass, one can compute the acceleration. Solving the differential equation numerically yields the current speed and position. The same basic model also describes other systems, especially in granular flow (e.g., Rao and Nott, 2008).

If the absolute values of the forces and the mass are not of interest in the simulation, the mass can be assumed to be 1. Therefore, I use $m = 1$ for simplicity, which leads to $\ddot{x} = f$. If $m \neq 1$, the forces have to be adjusted accordingly. Using realistic masses may be meaningful when additional physical forces are present or the forces themselves are of interest. If physical forces are measured in experiments, such as the pressure in a dense crowd, they rely on realistic values for the mass too.

For the force-based locomotion model, I assume that the preferred position after the next step x_p is known. Determining this step can be realised with the heuristic

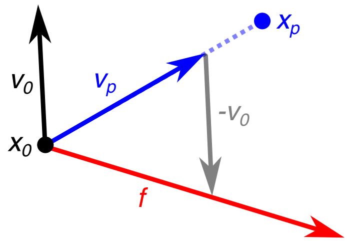

Figure 5.6: Illustration of the velocity and force vectors used in the force-based locomotion model. The current position is denoted by x_0 and the current velocity by v_0. The preferred next position and velocity are denoted by x_p and v_p, respectively. The force vector f is calculated by subtracting the current velocity from the preferred velocity. The preferred velocity is multiplied by the preferred speed (a scalar value), and the force vector is scaled by a factor that controls the reaction time. Equations 5.2 and 5.3 describe this calculation mathematically.

decision making presented in section 6.3 of the next chapter. Using the decision-making process from the optimal steps model would also be possible, but I did not implemented it for this work. The bodies of agents are represented as circles as in the optimal steps model.

Given the preferred position x_p and the current position x_0, a preferred velocity vector v_p is calculated. In order to reach the preferred position, the velocity vector has to point in the direction of x_p, and the preferred speed has to be known. At first, the velocity vector is normalised by its length $||x_p - x_0||$ and then multiplied with the preferred speed:

$$v_p = \frac{x_p - x_0}{||x_p - x_0||} \times \text{preferred speed}. \tag{5.2}$$

Formula 5.2 could already be used for a continuous velocity-based locomotion layer. For a force-based model, another step is necessary.

Acceleration represents the change in velocity and, for $m = 1$, equals the force. I use the difference in the current velocity to the preferred velocity and scale it by a factor. The resulting term is employed directly as the force applied to the agent. The formula is

$$f = (v_p - v_0) \times \text{scaling factor}. \tag{5.3}$$

The velocity and force vectors are illustrated in figure 5.6. In other force-based models, such as the social force model (Helbing and Molnár, 1995), the scaling factor is usually given by $1/\tau$, where τ is a relaxation time. The relaxation time controls how fast the changes in direction induced by the force are realised. For example, a small τ leads to greater forces and thus fast changes in direction. The same reasoning holds for the scaling factor in equation 5.3 but with inverted values:

$$\text{scaling factor} = \frac{1}{\tau}. \tag{5.4}$$

With known starting positions x_0 and velocities v_0 of agents, the system represents

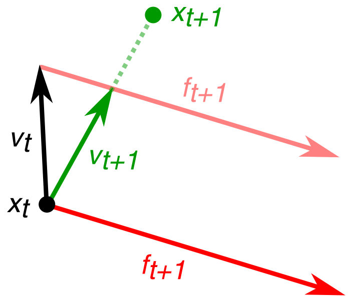

Figure 5.7: Illustration of the numerical solution of the force-based locomotion model (figure 5.6) based on the Euler method. The current position x_t and the current velocity v_t are used to compute the force f_{t+1}, which is applied for the next step $t+1$. The numerical scheme integrates the acceleration into the next velocity v_{t+1} and finally the next position x_{t+1}.

an initial value problem for differential equations:

$$\begin{array}{lcl} \dot{x} & = & v, \\ \dot{v} & = & f, \\ x(t_0) & = & x_0, \\ v(t_0) & = & v_0. \end{array} \tag{5.5}$$

The decision process is continuous, that is, the decision of where to go next is updated constantly. For the numerical solution, the decision is updated for every time step, and a new system of differential equations has to be solved. The desired position x_p may never actually be reached since it changes as the agent approaches it.

With the information of the current time step t, a step of the Euler method is applied to numerically compute the next time step $t+1$. For the numerical integration, the position and velocity vectors of the current time step t are used to compute the next velocity v_{t+1}, and the new velocity vector of $t+1$ is used to compute the next position x_{t+1}. The force $f(x_t, v_t)$ is calculated according to equations 5.2 and 5.3. Hence, the numerical step is defined by

$$v_{t+1} = v_t + f(x_t, v_t, x_{p,t}) \times \Delta t, \tag{5.6}$$

$$x_{t+1} = x_t + v_{t+1} \times \Delta t, \tag{5.7}$$

where Δt is the numerical time step length. The computation is illustrated in figure 5.7. Equation 5.6 describes an explicit Euler step and equation 5.7 an implicit Euler step. Therefore, taken together, a step of the semi-implicit Euler method is applied (e.g., Deuflhard, 1985).

The preferred position x_p is updated for every numerical time step t. Alternatively, one could update x_p and then use it until the agent has covered the distance predefined by the step length. In some initial tests, the latter method led to somewhat unrealistic detours when agents evaded others. Therefore, I use the continuous update. With parameter calibration and a more complex decision-making process that does not lead

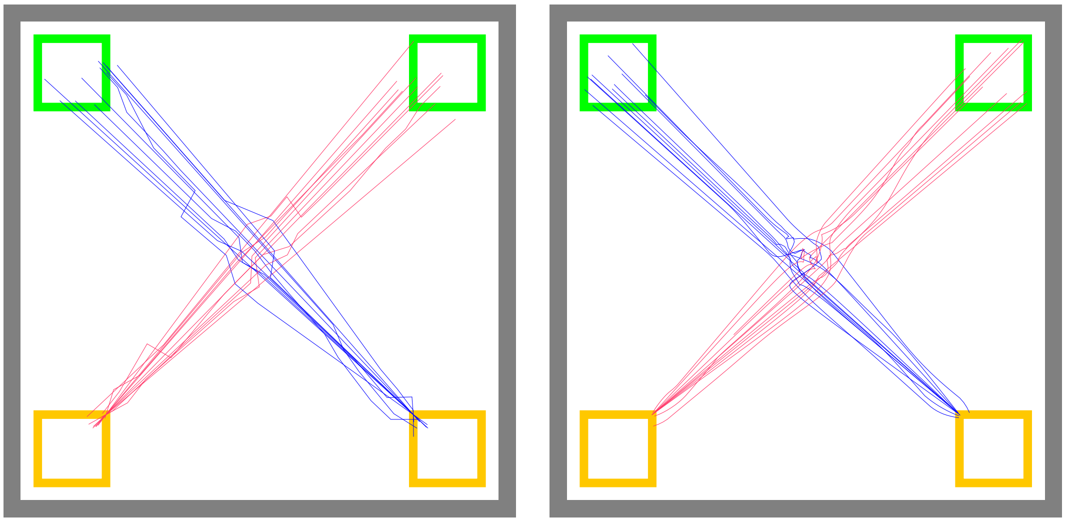

Figure 5.8: Simulated pedestrian trajectories for the comparison of the two locomotion models for pedestrians: discrete stepping process on the left and continuous force-based process on the right. For the decision making, the tangential evasion heuristic was used (subsection 6.3.1). Agents are created in the green rectangles on top, one agent every 10 seconds for the duration of 10 seconds for each source. They walk towards the yellow rectangle diagonally on the bottom. The whole area of the scenario has side lengths of 15 m. For the discrete stepping process (on the left), sudden changes in direction can be observed. In the continuous locomotion scheme, changes in direction are continuous.

to sudden changes of direction, the stepwise update of x_p may be an interesting concept to represent stepwise motion. Updating the decision only once for each movement step would also reduce the computational effort.

Other numerical solutions could be employed for the equation (e.g., Vesely, 2001), but the change of direction in the decision layer and the mechanism for collision prevention described in the following paragraphs lead to discontinuities, which complicate the use of higher-order solvers. With a fixed preferred position x_p for some time, higher-order solvers may be employed. In any case, the sudden stopping in case of a collision remains an issue.

Although the decision-making layer usually implements some kind of collision evasion algorithm, it does not ensure that collisions cannot happen on the physical layer. For example, an agent may choose to step in a direction that leads to a collision. Then the physical layer has to simulate the collision. As a basic model, agents in my approach simply cannot overlap, that is, they remain at the current position if the next numerical step would lead to an overlapping.

To realise this, I use a modification of the system described in equations 5.6 and 5.7. Instead of setting the next position directly, as in equation 5.7, x'_{t+1} is computed for all agents in the simulation first. Then collisions are checked with the updated positions x'_{t+1}. For all agents with a collision in x'_{t+1}, the next position x_{t+1} is set back to x_t. Equation 5.7 is replaced by

$$x'_{t+1} = x_t + \Delta t \times v_{t+1}, \tag{5.8}$$

$$x_{t+1} = \begin{cases} x_t & \text{if } x'_{t+1} \text{ leads to a collision} \\ x'_{t+1} & \text{else.} \end{cases} \tag{5.9}$$

The distinction of cases introduces discontinuities and hence complicates the use of higher-order numerical solutions. A mollification of the model may restore the requirements for higher-order numerical schemes (Köster et al., 2013).

The simple stopping of agents in the case of a collision is perhaps the most basic approach for collision resolution. A more advanced physical model would take into account the forces present in the case of a collision. One approach well-known in the simulation of granular flow is the discrete element method (DEM, Cundall and Strack, 1979; Kleinert et al., 2013), which can consider a variety of forces, such as friction and recoil. Given that the method described above is based on forces, it could easily be integrated with a DEM simulation.

Simulation results comparing the discrete stepping process and the force-based model are shown in figure 5.8. For the kinetic model of stepping, sudden changes of direction can be seen. In the continuous, force-based model, changes of direction over time are smooth because of inertia. The two physical processes can be combined with either local utility optimisation, heuristic decision making, or other models on the psychological layer (chapter 6).

5.4 Summary

In this chapter, I introduced the physical layer – the first of the three layers that, together, cover pedestrian dynamics. Although the physical layer is interconnected with the psychological layer, as an idealisation it is useful to separate the two. The psychological layer has to provide the decision of where to go next. The resulting position and possibly also the resulting velocity described in the physical layer is in turn used by the psychological layer.

I drew on the literature from biomechanics in section 5.1. Two perspectives on human movement can be taken: kinematics and kinetics. Kinematics describes movement without the consideration of forces, whereas in kinetics the various forces leading to movement are studied. A variety of simulation models has been proposed for the individual human walking process. Most models are situated in the coronal or sagittal plane. In contrast, pedestrian dynamics usually studies movement in the transverse plane. Moreover, biomechanical models of walking seem to be too complex for a simple representation in simulations of pedestrian dynamics. Nevertheless, there is a wealth of research that can be drawn on when further developing the physical layer.

Robotics (section 5.2) also has to study the mechanical process of movement. Especially walking robots can only be engineered with such knowledge. Findings in robotics can be of interest for the physical representation of agents in a simulation of pedestrian dynamics. The concept of dynamic walking seems especially promising because based on it, physical models (robots) have been built that exhibit a natural

		Decision making	
		Utility optimisation	Cognitive heuristics
Locomotion	Discrete steps	a	b
	Force-based	c	d

Table 5.1: Possible combinations of locomotion and decision making models for the respective two layers. Combination a corresponds to the optimal steps model, b to the simulation approach used in subsection 6.3.3, d to the simulations in subsection 5.3.2. The locomotion models presented in this chapter are highlighted in yellow.

walking behaviour. As was the case for biomechanics, the models are still to complex for a basic physical representation but may be of interest in future developments. I discuss other aspects of robotics such as decision making in the next chapter.

I presented two models for pedestrian locomotion in section 5.3. The first one is the discrete stepping process also used in the optimal steps model. It is a kinematic model and thus only simulates movement without a representation of forces. It is discrete in time and space because agents move forward instantly, but agents can move to arbitrary positions in space and at arbitrary points in time. Hence, agents act in continuous space and time. This model opens pedestrian simulation for a variety of studies, including the impact of crowd movement on the built environment. The model can be combined with other decision-making schemes, that is, models on the psychological layer such as cognitive heuristics.

The second model is force-based but has to be given the decision on where to step. In contrast to the social force model, it does not describe decision making with forces but only carries out the locomotion when the decision is already made. The model is a kinetic approach, which has the advantage that forces are modelled explicitly. It can be combined with other forces, such as contact forces.

The model on the locomotion layer has to be selected based on the objectives of the simulation. Table 5.1 summarises the options and highlights the two models for locomotion I discussed in this chapter. The two approaches provide a basis for a wide variety of studies. The discrete stepping process has the additional advantage of an implicit natural discretisation, which allows for efficient computation. The force-based approach seems to be more flexible in future developments. For example, the stepping process could also be used in a force-based model. Detailed models of biomechanics could be incorporated, which would lead to additional computational costs. Therefore, one has to weigh the advantages and disadvantages when selecting the respective models.

Chapter 6

The psychological layer: Heuristic decision making

The agents can carry out movement decisions given the locomotion layer introduced in the previous chapter. Models on the psychological layer generate the movement decisions (figure 6.1). They determine the position where the agent wants to step next within the proximity of the agent. In some cases, the locality of this decision is not that clearly defined, for example, when agents anticipate their motion some steps ahead.

The physical and the psychological layer cannot be separated entirely in the model as they are strongly linked in the real world too. For simulation, a separation is still convenient and justified. The physical layer simulates stepping behaviour, changes of direction through forces, the actual speed of movement, and so on using the decisions made by the psychological layer. The psychological layer may use the position and velocity of other agents as input. Some knowledge of the agent about the physical process can also be assumed, especially in real humans. For simplicity, this is largely ignored in the concepts I present but may be an interesting direction of future research.

The overall question that has to be addressed by the psychological layer is *where to step next*. This question may entail forward planning or, at least, some local strategic

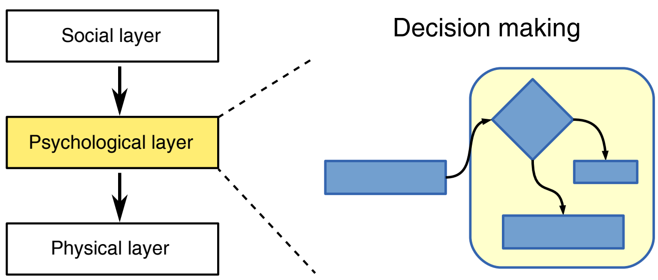

Figure 6.1: Illustration of the model separation into three layers. This chapter focuses on psychological aspects, builds on an existing physical layer, and provides a basis for the social layer.

component such as what distance the pedestrian wants to keep to other pedestrians and walls. Other motivational influences are what speed the pedestrians want to walk with and to which target they choose to go. The latter falls into the category of route choice or path planning, which I do not treat in detail here. Nevertheless, some target must be known in order to make the decision on where to move next. I assume that the target such as an exit door or any target area is provided by a route choice component (e.g., Kneidl, 2013).

Human behaviour is highly complex and diverse. For pedestrian dynamics, only those aspects of psychology are of interest that determine motion. Still, for a variety of reasons, only a small subset of psychological features can be represented in a model (section 1.3 in the introduction). Specifically, the psychological process has to be modelled in the simplest way so that it reproduces or describes observed phenomena.

Many relevant psychological differences can be found in pedestrians, including knowledge of the environment they are in, experience in similar situations, stress level, motivation, sense of security and safety, but also the ability to lead or the tendency to follow others. I discuss some social aspects and group behaviours in chapter 7. In the present chapter, many abilities and features of humans are excluded from the model. For example, agents in the simulation do not have psychological differences or memory and do not plan their movement and that of others forward explicitly. However, some of the concepts used for modelling decision making of pedestrians have implicit forward planning. Agents in the model only use local information, which is an approximation to the perception of humans. More elaborated theories like cognitive maps are not used. Perceptual constraints may be an interesting extension, especially for the approach with cognitive heuristics.

I first draw on related fields that study decision making, especially in the context of local movement. These are the behavioural sciences psychology and animal behaviour (section 6.1). I discuss the concept of bounded rationality and cognitive heuristics (subsection 6.1.3), which is the paradigm the model described in section 6.3 is based on. Section 6.2 is dedicated to the engineering aspects of decision making, namely artificial intelligence. In the rest of the chapter, I use this background and present a decision-making model for pedestrian dynamics (section 6.3). An important behaviour in pedestrian scenarios is waiting and remaining, which I cover in section 6.4.

6.1 Aspects from behavioural sciences

The psychological and the physical layer are interconnected (chapter 5) in the model as well as in reality. Decision making is an interaction with the physical world that goes in both directions (Raab et al., 2009; de Oliveira et al., 2009; Green and Heekeren, 2009; Schack and Ritter, 2009). For example, pedestrians visually perceive the environment and decide on what to do next on the psychological layer using the visual cues. This may lead to an action such as taking a step in a specific direction in the physical world. Decision making relates to the physical world on yet another level: mental activities can be observed physically in the body, that is, the nervous system. With the physical layer I propose here, I do not refer to the nervous system and mental activities, which are exclusively represented by the psychological layer.

Every choice that affects the movement of pedestrian crowds can be relevant for the decision-making layer. Hoogendoorn and Bovy (2004) define three levels of decision making for individual pedestrians – the strategic, tactical, and operational level:

> 1. Departure time choice, and activity pattern choice (strategic level);
> 2. Activity scheduling, activity area choice, and route-choice to reach activity areas (tactical level);
> 3. Walking behavior (operational level).
>
> (Hoogendoorn and Bovy, 2004)

The behaviour I study in the following would best be described as the operational level of this model. However, some decisions that affect the walking behaviour could be described as strategic too. For example, how much distance pedestrians keep to a wall or other pedestrians is also a strategic decision. Therefore, I use a different terminology: I refer to the decision models that describe walking behaviour as *local decision making* in contrast to route choice or path planning and global strategic decisions on the overall goal. Examples of decisions that I do not consider local are the choice of which exit to take, what kind of activity to follow, and so on.

The following subsections are dedicated to aspects from psychology, animal behaviour, and decision making. I discuss the literature as background for simulation models of pedestrian dynamics.

6.1.1 Aspects from psychology

Psychology is the "scientific study of the behavior of individuals and their mental processes" (Gerrig and Zimbardo, 2002). Many aspects of psychology may have an impact on pedestrian and crowd dynamics. For the study of human behaviour in a computer simulation, a minimal set of aspects has to be selected that describes relevant phenomena. Areas that are of interest for the simulation of individual pedestrians are perception and behaviour, motivation, cognition, and social psychology. In the following, I report definitions for the terms and draw on the literature for relevant background. I start by outlining the fields that study the interfaces to the physical world: perception and behaviour.

Perception describes the "processes that organize information in the sensory image and interpret it as having been produced by properties of objects or events in the external, three-dimensional world" (Gerrig and Zimbardo, 2002). This definition first says that perception collects information from the physical world and integrates it into a mental representation of it. Furthermore, perception deals with the interpretation of the input as it represents the information in a coherent picture of the physical world. Perception is of great importance for pedestrian dynamics because pedestrians interact with their environment based on the information they have on it. The current behaviour is also determined by other aspects, including memory and learned behaviour. While these are relatively constant, perception makes the behaviour adaptive for different situations and contexts.

Important perceptual capacities of pedestrians are the estimation of distances (e.g., Ziemer et al., 2009; Iosa et al., 2012) and motion (e.g., Johansson, 1973; Albright and Stoner, 1995). These abilities allow us to navigate through a fast changing environment in many different contexts. Without them – for example due to vision impairment or because of the lighting conditions – we can draw on aural and tactile cues. Nevertheless, in most cases, impairment of vision also constraints our navigational abilities. In complex environments, we may not be able to process all available information and then have to focus our attention on relevant features such as possible collisions with others (Jovancevic et al., 2006). It is important to note that perception can be highly selective and even blend out events that seem hard to ignore (Hastorf and Cantril, 1954).

On the other side of the interaction with the physical world stands behaviour. It is defined as the "actions by which an organism adjusts to its environment" (Gerrig and Zimbardo, 2002). The mental process can often only be studied indirectly through behaviour, which is the process we can observe directly. General findings on behaviour in psychology can be of interest for the development of pedestrian simulation models.

At the core of all human action stands motivation, the "process of starting, directing, and maintaining physical and psychological activities; includes mechanisms involved in preferences for one activity over another and the vigor and persistence of responses" (Gerrig and Zimbardo, 2002). A rough classification of behaviours can be undertaken based on the distinction between approach and avoidance motivation:

> Both approach and avoidance motivation are part of our evolutionary heritage, and we certainly cannot survive, either physically or psychologically, without both types of motivation. Certain tasks in negotiating the environment and our social world require avoidance motivation, and avoidance motivation is undoubtedly adaptive in some instances. For example, it is imperative that our perceptual system be perpetually vigilant for physical danger or it is likely that our lifespan would be greatly truncated . . . (Elliot, 2006)

The scalar fields (section 3.5, chapter 3) used in many models of pedestrian dynamics can be interpreted as a representation of approach and avoidance motivation. For example, high potentials in the social force model lead to an avoidance of these areas. However, approach and avoidance motivation is a simple perspective that cannot explain pedestrian behaviour in depth.

Pedestrians usually follow motivations that lead to specific behaviours: avoiding collisions, reaching a target, keeping a certain distance to others, walking with a preferred speed, staying with a group, communicating with a group, staying in a safe area, and so on. The specific motivations may not seem important for a simulation model, but they give the necessary background and can be used to clearly separate aspects of the model. Listing the motivations may also help to make explicit what a given model does or does not capture and to identify shortcomings of existing models.

Cognitive psychology is the "study of higher mental processes such as attention, language use, memory, perception, problem solving, and thinking" (Gerrig and Zimbardo, 2002), thus, it comprises perception as well as decision making. Knowledge of general cognitive abilities and paradigms help to develop plausible models of human behaviour. Subsection 6.1.3 is dedicated to the theory of bounded rationality (Simon, 1990) and cognitive heuristics (Gigerenzer et al., 1999) as a paradigm, which I also use for the model of pedestrian dynamics in section 6.3.

Social psychology is the "branch of psychology that studies the effect of social variables on individual behavior, attitudes, perceptions, and motives; also studies group and intergroup phenomena" (Gerrig and Zimbardo, 2002). The study of pedestrian dynamics is also a study of social interactions, and social psychology may provide important findings for it. In this work, I separate group behaviour and explicitly social behaviour from proximity navigation. This separation is not entirely successful because interacting agents already describe social behaviour, even if they only try to navigate around others. The goal is to provide a basic layer for pedestrian navigation in this chapter that can be extended with models of collective and group behaviour in the next chapter.

Simulation models of pedestrian dynamics are often advertised for the study of emergency evacuation scenarios (e.g., Gwynne et al., 1999; Pan et al., 2007). Sometimes the term "panic" is used (e.g., Helbing et al., 2000a) to describe human group behaviour in emergency situations that led to a loss of lives. Since the term "panic" suggests irrational behaviour, it has been dismissed in the scientific literature (e.g., Johnson, 1987; Aguirre, 2005; Drury et al., 2013). Nevertheless, human behaviour is likely to change in emergency situations because of a variety of factors, including additional stress, novelty of the situation, impairment of perception, and the urgency, which pressures individuals to make decisions fast. Behaviour that may seem irrational from an outside observer or in retrospect may not be all that irrational when considering the circumstances people were in.

Many other topics in psychology can further the understanding of pedestrian behaviour in a variety of contexts and from different perspectives. For a concrete simulation approach, one has to select relevant aspects, otherwise, the model becomes too complex for validation. The argument of parsimony becomes even more important when trying to incorporate theories from social psychology such as the social identity model (section 7.1).

6.1.2 Aspects from animal behaviour

A field related to psychology is animal behaviour (e.g., Manning and Dawkins, 2012). In animal behaviour, topics like cognition and decision making are studied, too. On the one hand, there are similarities between humans and other species – but one also has to make sure not to neglect the differences. In general, humans could be regarded to have higher cognitive abilities. On the other hand, some species may supersede the abilities of humans when faced with certain situations, especially as the physiology of a species gives them specific abilities.

Most phenomena in pedestrian dynamics can be described as self-organisation, that is, individuals behave in a way that yields the phenomena without a superordinate individual controlling or instructing the process. Self-organisation is well-studied in biology (Couzin and Krause, 2003). An important concept that cannot be separated clearly from self-organisation and may in fact often cover the same topic is collective behaviour and collective motion (e.g., Sumpter, 2006 and Vicsek and Zafeiris, 2012). In these areas, similarities among swarms, flocks, and also pedestrian crowds can be found (Moussaïd et al., 2009a).

Another direction of research treats animal navigation (e.g., Collett and Graham,

2004. In navigation, mainly individual movements over longer distances are studied and not so much the interaction and collective motion of groups. Concepts, such as path integrations have given insights into how animals forage and find their way back home (e.g., Etienne et al., 1998; McNaughton et al., 2006). Long-range travels are studied in the migration of animals (e.g., Dingle and Drake, 2007; Dingle, 2014).

Findings in animal behaviour may contribute to the study of pedestrian dynamics. An example are cognitive or perceptual constraints in the number of neighbouring individuals that are taken into account for movement decisions (Ballerini et al., 2008; Bode et al., 2010; Strandburg-Peshkin et al., 2013). Moreover, human behaviour is also studied by researchers in the domain of animal behaviour and may highlight similarities among species.

In collective animal behaviour, mostly swarms and flocks are studied. Here, animals form a collective for a specific purpose such as protection against predators. Pedestrians usually form crowds for other purposes or even without a purpose in a physical crowd. I give a definition of psychological and physical crowds in the next chapter, section 7.1. Therefore, models from biology cannot be used directly for human behaviour. When using models from animal navigation, a similar line of argument can be made: animals typically set out to look for food sources. This is not the case for humans in a modern environment. Some basic models and mechanisms may still be of use for the study of pedestrian behaviour, including cognitive maps and path integration (Mittelstaedt and Mittelstaedt, 1980; Collett and Graham, 2004; McNaughton et al., 2006). However, these would mainly be of interest in route choice models, which are not the focus of my work.

6.1.3 Decision making and bounded rationality

Decision making is defined as the "process of choosing between alternatives; selecting or rejecting available options" (Gerrig and Zimbardo, 2002). There are descriptive and normative models of decision making. Descriptive models aim at a better understanding of natural processes such as animal behaviour. Normative models, on the other hand, try to aid decision making. For example, a formally defined decision process my help medical doctors to efficiently act in emergency situations (Breiman et al., 1984, pp. 1–2).

In computational science, it is of great importance that the two not be confused. When simulating pedestrian dynamics, one tries to understand, reproduce, and predict human behaviour. In some cases, a what-if scenario may require looking for normative solutions such as the question "what behaviour might improve the outcome?" If that is the case, it has to be clearly separated from the description of human behaviour. Although the "double role—to describe and to prescribe—does not map easily onto a sharp divide between descriptive and normative models" (Gigerenzer and Selten, 2001, p. 1), I argue that the confusion of the two can lead to a severe misrepresentation of human behaviour and ultimately to misguided model predictions.

Decision making is studied in behavioural sciences but also in other fields, including economics (Simon, 1959). Traditional models in economics assumed unbounded rationality, which means that agents act in a way that allows them to reach optimality according to some criteria (e.g., Gigerenzer and Todd, 1999). To come by such a solution, they may draw on unlimited resources for reasoning. This concept has been criticised by Simon (1955), who later coined the term *bounded rationality* as an alternative approach for the understanding of human decision making (e.g., Simon, 1990). Bounded rationality suggests that – in contrast to traditional models – resources for reasoning are limited.

> Humans and animals make inferences about their world with limited time, knowledge, and computational power. In contrast, many models of rational inference view the mind as if it were a supernatural being possessing demonic powers of reason, boundless knowledge, and all of eternity with which to make decisions. Such visions of rationality often conflict with reality. (Gigerenzer et al., 1999, p. 7)

The proponents of bounded rationality say that it specifically "dispenses with optimization, and, for the most part, with calculation of probabilities and utility as well" (Gigerenzer and Selten, 2001, p. 3) as they argue that optimisation is often well out of the boundaries of our cognitive capacities.

As an important example of a model of bounded rationality, I briefly describe aspiration theory (Starbuck, 1963). Aspiration theory presumes that individuals try to reach a certain level of satisfaction rather than optimality. For example, you may not continue looking for a vendor of a product after you have found one that offers the product at a price you deem favourably. In contrast, according to unbounded rationality, you *would* continue consulting *all* vendors of the product and select the one that offers it at the lowest price. In reality, this search may not even be feasible to finish in one lifetime.

Tversky and Kahneman (1974) conducted a series of experiments that show how, under certain conditions, humans tend to make biased decisions that seem to be inconsistent and irrational. For example, they demonstrated the effect of framing (Tversky and Kahneman, 1981). In an experiment, they present two mathematically identical problems to participants. The participants have to decide which of two options they would choose. Both option A and B have the same expected value, but option B has some variance while option B does not. If the problem is described in terms of gain, participants prefer option A, which is not associated with uncertainty. If the problem is described in terms of loss, they tend to choose option B, which is subject to variance. This leads to the authors' statement that "choices involving gains are often risk averse and choices involving losses are often risk taking" (Tversky and Kahneman, 1981). The decisions seem to be highly inconsistent because, formally, both problems are exactly the same, yet participants decided differently because of the framing. Rationality would seem to demand the same decision independently of the framing. This line of research is also often described as the heuristics and biases school (e.g., Kelman, 2011).

Gigerenzer and Goldstein (1996), on the other hand, demonstrated that models of bounded rationality can even outperform the traditional model of rational inference. Heuristics in this context are algorithms that allow for fast and efficient computation and do not necessarily perform worse than optimisation in real world scenarios. Gigerenzer and Todd (1999, pp. 25–29) give a detailed definition of the term *heuristic*. This line of research led to the fast and frugal heuristics school, which argues that, in general, bounded rationality works well under many circumstances (Gigerenzer et al., 1999; Todd and Gigerenzer, 2000; Gigerenzer, 2008).

Goldstein and Gigerenzer (2002) propose to design and test cognitive heuristics that are "(a) ecologically rational (i.e., they exploit structures of information in the environment), (b) founded in evolved psychological capacities such as memory and the perceptual system, (c) fast, frugal, and simple enough to operate effectively when time, knowledge, and computational might are limited, (d) precise enough to be modeled computationally, and (e) powerful enough to model both good and poor reasoning". Particularly, complexity arises out of the behaviour of individuals in interaction with the environment, but the behaviour itself need not be complex (Simon, 1996).

The two perspective on cognitive heuristics – the heuristics and biases school and the fast and frugal heuristics school – have led to some dispute between their proponents (Kahneman and Tversky, 1996; Gigerenzer, 1996). Nevertheless, there is agreement on the fundamental principle of cognitive heuristics: "At some level of generality, there is widespread agreement that people are employing heuristics whenever they make a judgment or reach a decision without making use of some information that could be relevant or some computational abilities that at least some people possess" (Kelman, 2011, p. 3).

In pedestrian dynamics, one has to model both good and poor decision making. In most cases, pedestrian behaviour is highly efficient: accidents are the exception and we usually get where we want to go. This does not mean that it is necessarily optimal according to whatever criterion. Accidents do occur and probably everyone has experienced some kind of collision with other pedestrians. This agrees with the programme Goldstein and Gigerenzer (2002) propose (above, point c). The algorithms used in simulation innately have to be precise enough to be modelled computationally (point d). The other requirements of their programme are in line with the goal of a plausible decision-making model, especially that it be based on bounded rationality (point c) and founded in evolved psychological capacities (point b). Therefore, I consider this paradigm suitable for model development in pedestrian dynamics.

In biology, a similar direction can be found where behaviour is described by "rules of thumb". Hutchinson and Gigerenzer (2005) outline both cognitive heuristics and rules of thumb in biology and discuss similarities among the two. Conlin (2009) reviews heuristic decision making for navigation in humans. McLeod and Dienes (1996) describe how local movement decisions can be made using simple heuristics: fielders who try to catch a ball, instead of calculating the trajectory, only estimate whether the ball will land before or behind them and adjust their position accordingly over time. This example also demonstrates how it may seem as if humans computed the optimal solution but, in fact, can use a simple heuristic that produces a similar result.

The "adaptive toolbox" (Gigerenzer and Todd, 1999) of behavioural heuristics has several advantages that are also beneficial to the modelling of navigation decisions. First, all computations employed for the heuristics have to be based in capacities the considered species possesses, which makes them a plausible representation of real cognitive processes. Second, due to its simplicity, the decision process facilitates the understanding, modelling, and communication across disciplines without in-depth knowledge of the underlying mathematics and computations. Third, heuristics can be implemented and tested as hypotheses without necessarily affecting other heuristics of the same toolbox. Fourth, due to the flexibility, the resulting model can be expanded to capture further details or additional aspects of behaviour based on the current state of the model development. Fifth, the model parameters represent entities that are observable in nature, such as the distance swarming fish try to keep to each other or the number of other pedestrians considered for collision avoidance, in contrast to abstract concepts such as social forces or utility functions. Sixth, the implementation of behavioural heuristics could be automated with code generation facilitating the creation and testing of new hypotheses. For example, predefined heuristic building blocks could be offered and combined in a graphical interface and the code carrying out the operations generated automatically.

One could argue that other models of pedestrian dynamics such as the social force model (Helbing and Molnár, 1995) are already simple enough. However, the correct implementation of the social force model can still be challenging in spite of its mathematical simplicity (Köster et al., 2013). Furthermore, they may seem simple and intuitive to mathematicians and physicists, who are familiar with the mathematical formalism, but also may not to biologists or psychologists. Thus, I argue that cognitive heuristics are more intuitive, especially for researchers in the fields of behavioural sciences. The formal description of cognitive heuristics is not complex, even if the concrete algorithmic implementation can be. Behavioural scientists can still propose cognitive heuristics with little knowledge of mathematics and computer science because they do not have to know how to implement them.

In contrast to some models of pedestrian dynamics, cognitive heuristics do not optimise in the sense that they necessarily find the optimal solution. However, they still optimise the decision in the sense that they strive to find the optimum – which they may or may not find. Cognitive heuristics aim to find a “good-enough” solution (e.g., Gigerenzer, 2008). The optimal steps model (Seitz and Köster, 2012) contradicts this paradigm as it employs numerical optimisation of a utility function to determine the next step.

Having decided that cognitive heuristics are a suitable paradigm for the simulation of pedestrian dynamics, the question arises of how to practically apply this idea. The fast and frugal heuristics school suggest the following research agenda:

> The program of studying boundedly rational heuristics involves (a) designing computational models of candidate simple heuristics, (b) analyzing the environmental structures in which they perform well, (c) testing their performance in real-world environments, and (d) determining whether and when people really use these heuristics. The results of the investigatory stages (b), (c), and (d) can be used to inform the initial theorizing of stage (a). (Gigerenzer and Todd, 1999, p. 16)

Steps *a*, *b*, and *c* are covered in subsections 6.3.1 and 6.3.3 for a model I propose for the simulation of pedestrian dynamics, and I discuss step *d* in subsection 6.3.4.

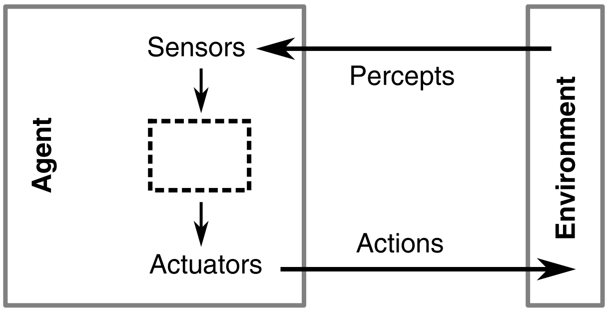

Figure 6.2: Schematic representation of an agent in artificial intelligence. The agent perceives the environment through sensors and acts in the environment through actuators. The empty box between sensors and actuators may be understood as the cognition of the agent. (Figure: adapted from Russell and Norvig, 2010, p. 35)

6.2 Aspects from artificial intelligence

The behavioural sciences study decision making descriptively. When building a machine or programme that makes some kind of decision, on the other hand, the focus is on engineering. In this section, I review literature from artificial intelligence, which can be considered to comprise the engineering of decision making. As in robotics (section 5.2, chapter 5), the objectives in engineering can deviate greatly from scientific modelling. In engineering, one might just want to build a machine that works well for some task but may not be a good model of natural phenomena. Nevertheless, for a computer simulation of human behaviour, one still has to design algorithms and implement software that describe behaviour, which are also core disciplines of artificial intelligence. Therefore, artificial intelligence can be an important resource for technologies that help to develop models of natural phenomena. This is also expressed by the early work on cybernetics by Wiener (1961), which touches on many topics at the interface between biological systems and artificial intelligence.

A common concept in artificial intelligence are agents: "An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators" (Russell and Norvig, 2010, p. 34). This model has parallels to the disciplines of psychology I discussed in subsection 6.1.1. The sensory organs and the perception can be seen as the sensors and the behaviour can be described by the actuators and actions of an agent. The basic agent model is shown in figure 6.2. The empty box between sensors and actuators may be understood as the cognition of the agent.

A variety of agents with different capacities can be defined given the basic agent structure where the agents differ in what happens within the dashed box (Russell and Norvig, 2010, chapter 2). For example, the reflex agent creates an internal representation of the environment and decides on an action following condition-action rules. Other types are model-based, goal-based, utility-based, and learning agents. This classification can be employed for microscopic pedestrian stream models, especially the complexity assessment of simulation approaches. The optimal steps model (chapter 4) can be described as a utility-based agent model. Agents in the optimal steps model perceive the positions of others around them. Then they define a set of possible actions, which are given by the possible next steps on the step circle. They choose one of the positions by assessing the respective utilities. Finally, they move to the position with the highest utility value.

An interacting collection of agents is a multi-agent system (e.g., Shoham and Leyton-Brown, 2008). The game-theoretic foundations can be of interest for pedestrian dynamics and describe some natural phenomena of self-organisation. However, a general multi-agent system may also be designed to fulfil a specific task in the spirit of engineering an efficient machine and hence does not necessarily describe natural phenomena.

Another field of artificial intelligence that is important for pedestrian dynamics is planning (e.g., Russell and Norvig, 2010, chapters 10 and 11). Especially for route choice, methods such as Dijkstra's algorithm (Dijkstra, 1959) and Sethian's fast marching algorithm (Sethian, 1996, 1999) can be employed to simulate the behaviour of pedestrians. An alternative are graph-based approaches (Kneidl, 2013), which are studied in artificial intelligence, too. However, these algorithms are on the level of animal navigation, that is, route choice behaviour, which is not the focus of my work.

In robotics, artificial intelligence is applied for the control of autonomous robots (e.g., Arkin, 1998, and Russell and Norvig, 2010, chapter 25). Topics such as route choice and path planning have an application for robots that must actively navigate through an environment with obstacles. Another area is the avoidance of collisions with moving obstacles, which can be solved using the velocity-obstacle approach (Canny and Reif, 1987; Fiorini and Shiller, 1998). Especially interesting is the development of control mechanisms in robots by Brooks (1989), who designed control mechanisms in a modular way that can be extended and combined. The approach is somewhat similar to the heuristics toolbox proposed by Gigerenzer and Todd (1999).

For the representation of heuristic building blocks, some of the formalisms used in artificial intelligence may be helpful. For example, "statecharts, extend conventional state-transition diagrams with essentially three elements, dealing, respectively, with the notions of hierarchy, concurrency and communication" (Harel, 1987). Statecharts have some commonalities with Venn diagrams and can be defined as an application of higraphs (Harel, 1988). Statecharts also graphically describe finite state machines (e.g., Kohavi and Jha, 2010), a concept that is suitable for the description of agent behaviour. Extensions of state machines, including hierarchical finite state machines (Alur et al., 1999), allow for the structuring of complex control mechanisms and can still be analysed formally (Alur and Yannakakis, 1998). The application of hierarchical finite state machines for the definition of agent behaviour in pedestrian simulations was proposed by Kielar et al. (2014) and Kielar and Borrmann (2016).

In the animation of avatars, swarms, flocks, herds, and schools (Reeves, 1983; Reynolds, 1987), simulation methods are used that can be of interest for scientific modelling. Especially as in animation a certain degree of realism is also desired, and hence, the goal of reproducing phenomena from nature is given in principle. Artificial intelligence and animation are commonly used in the development of computer games (Millington and Funge, 2009). For example, Reynolds' steering behaviours (Reynolds,

1999) have been used to simulate pedestrian behaviour. Reynolds (1987) also describes behavioural rules that somewhat resemble heuristic decision making or rules of thumb:

1. Collision Avoidance: avoid collisions with nearby flockmates
2. Velocity Matching: attempt to match velocity with nearbyflockmates
3. Flock Centering: attempt to stay close to nearby flockmates

(Reynolds, 1987)

The animation of crowds is covered in the textbooks by Pelechano et al. (2008) and Thalmann and Musse (2012).

The representation, classification, and development of agents seem to be suitable for simulation models in pedestrian dynamics. Software frameworks for intelligent agents could be employed for the simulation of pedestrian behaviour. Furthermore, the formal study of multi-agent systems may provide insights that reflect natural phenomena. In the control of robots, a variety of methods is applied that are also relevant to pedestrian simulation, such as route selection and motion planning. Statecharts and finite state machines seem useful for the description of heuristic building blocks, especially in the future when the described behaviours become more complex. In game development and animation some goals are even the same as in scientific modelling since the simulated behaviour must often, at least, look natural, but entertainment is usually the primary goal. Other specific fields of artificial intelligence such as cognitive sciences (e.g., Stenning and van Lambalgen, 2008) may also provide insights for the study of pedestrian behaviour.

In summary, there are differences between scientific model development and artificial intelligence. Namely, the objective of artificial intelligence is engineering and hence not necessarily the description of natural phenomena. Nevertheless, technologies developed in artificial intelligence can also be used for the development of computer simulations in scientific studies. For example, I use the fast marching algorithm and the agent perspective in this work.

6.3 Decision-making models for pedestrians

In this section, decision-making is understood as part of the individual psychological layer, and both locomotion and group behaviour are not part of it. I point out how the decision-making process works together with the locomotion layers presented in chapter 5. Every microscopic simulation model of pedestrian dynamics (chapter 3) has some kind of decision-making process for individual agents, although sometimes it is not obvious at first how decision making is represented. In force-based models (section 3.3), decisions are implicitly determined through the potential field that accelerates agents. In velocity-based models (section 3.2), decision making is encoded in the velocity equation. For cellular automata (section 3.1), different decision-making approaches exist. While some can be interpreted as utility optimisation (e.g., Gipps and Marksjö, 1985), a common approach is a probabilistic model that only describes behaviour stochastically (e.g., Burstedde et al., 2001). Although the movement of agents from cell to cell can be seen as following rules, the rules are often complex and changing over time or describe behaviour stochastically. Another class of cellular automata determines agent behaviour through simple rules (e.g., Blue et al., 1997; Fukui and Ishibashi, 1999), which is somewhat similar to the idea of heuristics but without the justification from psychology.

The optimal steps model (chapter 4) is a complete simulation model of pedestrian dynamics in the sense that it represents decision making as well as locomotion. The next position is chosen based on the scalar field, which is best interpreted as utility. Therefore, the decision-making paradigm in the optimal steps model is local utility optimisation. The idea of utility optimisation has been criticised as an implausible representation of human decision making by Simon (1990) and has later been further challenged by Gigerenzer et al. (1999).

In this part of my work, I try to find simulation models of pedestrian dynamics that not only exhibit emergent phenomena of real crowds but at the same time represent a plausible model of the underlying processes. In subsection 6.1.3, I argued that bounded rationality and cognitive heuristics provide a paradigm that matches the requirements of a plausible natural process and has the necessary precision for computer simulation. Moussaïd et al. (2011) proposed to use simple rules for the simulation of pedestrian dynamics and also referred to "behavioural heuristics" citing Gigerenzer et al. (1999). They also separated the decision-making from the physical layer using a locomotion process based on the social force model. However, the decision-making process the authors suggested leads to an optimisation problem that makes it computationally demanding. This decision-making model may describe some aspect of the real process, but I argue that pedestrians make decisions following simple rules that are computationally less demanding.

A variety of models, including the social force model, was able to reproduce crowd phenomena on the basis of concise equations. In terms of simplicity, this may seem a desirable goal. However, the calibration and the extension with more complex behaviours are still challenging (e.g., Johansson et al., 2007; Moussaïd et al., 2009b). Furthermore, there seems to be a limitation in the number of behaviours that can be introduced to the model and the manageability of parameters (Gigerenzer et al., 1999). With cognitive heuristics, on the other hand, new behaviours can be introduced easily and complement or make use of the existing ones. I argue that, for complex behaviours, cognitive heuristics are better understandable and easier to handle than previous models.

Most models discussed in chapter 3 aimed at reproducing a variety of behaviours with one decision scheme. In contrast, cognitive heuristics "are neither particular nor general, but they have some intermediate range of applicability" Gigerenzer (2008). In addition to the computation of the heuristic, another task arises: the selection of a heuristic based on the context. It may seem as if the complexity was only shifted away to this problem. However, this step can also be realised with a heuristic scheme, and thus, simplicity is preserved. The heuristics models presented in the following subsection should be seen as hypotheses. The advantage of this modelling technique is that some behaviours can be tested and falsified without necessarily affecting the remaining toolbox. The approach provides a framework for modelling pedestrian behaviour that can be constantly improved.

6.3.1 Cognitive heuristics

> Optimization is an attractive fiction; it is mathematically elegant, and one can draw on a well-developed calculus. Compared to the beauty of optimization, the actual proximal mechanisms of humans and animals resemble the tools of a backwoods mechanic. The pleasing ideal of a universal calculus may have distracted researchers in many fields from exploring the contents of the adaptive toolbox. However, there is also another sense of beauty: the aesthetics of simplicity. There is a sense of wonder in how simplicity can produce robustness and accuracy in an overwhelmingly complex world. (Gigerenzer and Selten, 2001, p. 11)

Developing a pedestrian stream simulation according to the paradigm of cognitive heuristics may seem particular at first because many studies into heuristic decision making are based on pen and paper tasks (Gigerenzer et al., 1999). An example of heuristic movement decision making was given by McLeod and Dienes (1996), who studied how fielders catch a ball. Conlin (2009) reviewed heuristic decision making in the context of navigation. Blue et al. 1997 and Fukui and Ishibashi 1999 used simple rules – which are somewhat similar to the heuristics I describe in this section – to determine agent behaviour in a cellular automaton. Moussaïd et al. (2011) proposed a simulation with simple rules for the decision-making layer that leads to a rather demanding optimisation problem. Xu et al. (2012) used a similar line of argument for cognitive heuristics referring to literature by Gigerenzer and colleagues. Their model, however, makes use of numerical optimisation, too. No simulation approach to pedestrian dynamics is known to me that explicitly uses heuristic decision making without numerical optimisation.

Gipps and Marksjö (1985) argued that pedestrians integrate all information, which poses a problem too demanding for simulation in a computer. In contrast, I argue with Gigerenzer et al. (1999) that pedestrians do not use all available information but only need specific bits of it that are necessary for the next decision. Although not many studies on heuristic decision making in pedestrian behaviour can be found, the paradigm seems particularly apt for the description of pedestrian behaviour. As pedestrians we do not have much time to make a decision. We do not think it through or consult "system two", as Kahneman (2012) calls the slower but more thorough cognitive process.

> ...we consider limited time and limited knowledge as constraints under which people have already developed or learned their smart heuristics. This implies that an individual's repertoire of strategies includes some that take the constraints into account. We do not assume that a trade-off between effort and accuracy or an evaluation of strategies is computed during the decision process. Based on an individual's prior experience of decision making, a particular situation could prompt her or him to use a particular decision strategy. Rieskamp and Hoffrage (1999, p. 147)

A computational model of heuristic decision making can be decomposed into three parts: "guiding search for alternatives, information, or both, stopping that search, and

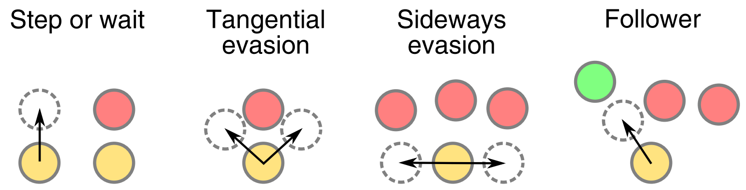

Figure 6.3: Illustration of microscopic emergent behaviours by agents following the heuristic-decision-making model. The focal agent is shown in yellow and other agents blocking the path are red. Dashed circles represent possible next positions of the focal agent. The green circle in the follower heuristic represents an agent walking in the same direction. (Figure: Seitz et al., 2015a)

making a decision" (Gigerenzer and Todd, 1999, p. 16). "Principles of search direct how information is searched, principles of stopping define when search is terminated, and principles of decision specify how the information searched for is used to make a decision" (Rieskamp and Hoffrage, 1999, pp. 143–144). "The principles of search, stopping, and decision are connected to each other. For example, when a heuristic searches for only one (discriminating) cue, this constrains the possible decision rules to those that do not integrate information" (Rieskamp and Hoffrage, 1999, p. 144).

In this section, I present a pedestrian simulation model based on the paradigm of cognitive heuristics[1]. The agents in this simulation only use information from their proximity. They do not explicitly plan ahead, but the heuristics can be seen as implicitly doing so. This is in line with the notion that heuristics encode efficient decision making without explicitly optimising for it. Why use complex planning when a simple heuristic also works? It seems more likely that humans and animals employ a simple cognitive process instead of a complex one if they both show the same results. Therefore, I argue it is unlikely that we use optimisation and explicit planning in proximity navigation.

Figure 6.3 illustrates the emergent effects of the heuristics I describe in the following (Seitz et al., 2015a). The step or wait heuristic makes agents approach the target if the path is free and remain at the current position if it is not. The tangential evasion heuristic makes agents evade tangentially if another agent is blocking the path. The sideways evasion heuristic additionally makes agents evade to the sides if tangential evasion is not possible. The follower heuristic lets agents choose another agent walking in the same direction to follow if the direct path to the target is blocked by oncoming agents.

The heuristic decision-making model can be combined with two locomotion models. The first one is the discrete stepping process, and the second one is the continuous force-based process that utilises the next step as target direction (section 5.3). For the following discussion in this subsection and the simulation studies in subsection 6.3.3, I use the discrete stepping process. A coherent description of the stepping process in combination with the heuristics can also be found in our paper describing the simulation model (Seitz et al., 2015a).

[1] Most of the ideas in the argumentation, heuristics model, simulation results (subsection 6.3.3), and future directions (subsection 6.3.4) are the result of collaborative work (Seitz et al., 2015a).

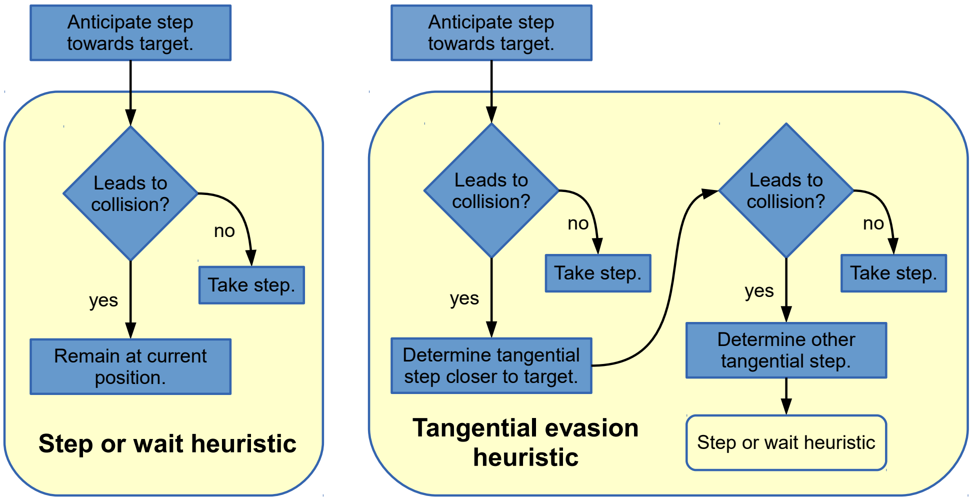

Figure 6.4: Flow charts of heuristics for pedestrian navigation in the proximity. Blue rectangles represent computational steps or actions, which may also include information search. Blue diamonds represent binary decisions. Yellow boxes with round corners are whole heuristic building blocks that can also be used within other heuristics. The step or wait heuristic makes agents take steps towards the target or, if the position is not free, remain at the current position. The tangential evasion heuristic lets agents try to evade collisions tangentially to the side. If tangential evasion is not possible, the step or wait heuristic is used, that is, the agent remains at the current position. (Figure: Seitz et al., 2015a)

In the simulation, agents are assigned a target at first. When they reach the target, they may either go on to the next target or are removed from the scenario if it is the final target. The target must allow for direct line of sight: no static obstacles on the path to it are encountered. This may not be the case in real world scenarios, but then a visibility graph with intermediate targets can be constructed that provides a direct line of sight between them (Kneidl et al., 2012; Kneidl, 2013). On the way to the target, agents have to navigate around others, which is described by the heuristic decision rules.

Figure 6.4 shows the first two, most basic heuristics for navigation in the proximity. The step or wait heuristic makes agents approach the target directly unless they encounter another agent who occupies a position overlapping with the position of the next step. In the latter case, agents simply remain at the current position. This behaviour produces an emergent effect similar to queueing because agents do not overtake one another. If all agents approach the same target, no permanent jams prohibit egress to the target. In contra-flow scenarios, however, two agents moving in opposite directions and facing each other will not evade and hence stall.

The tangential evasion heuristic (shown on the right in figure 6.4) allows agents to resolve such situations. If a collision is detected for the next step, agents evade tangentially to the side. Therefore, individual encounters of agents moving in opposite direction do not lead to stalling. When all agents approach the same target, the emergent behaviour seems slightly more competitive as agents tend to evade peers in

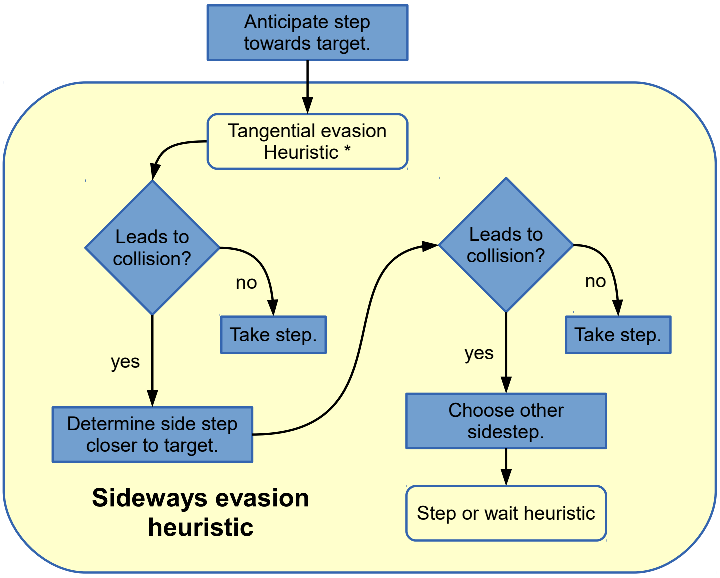

Figure 6.5: Flow chart for the sideways evasion heuristic. At first, agents following this heuristic try to use the tangential evasion heuristic to navigate towards the target. If that fails, they try to evade to the side. They resort to the step or wait heuristic if the positions to their sides are unavailable too. (Figure: Seitz et al., 2015a)

front of them although they may not actually be able to overtake. In dense crowds with different walking directions, agents still block each other and jamming occurs.

To allow for additional flexibility in contra-flow scenarios, the sideways evasion heuristic introduces steps to the side if tangential evasion is not possible. This decision process is shown in figure 6.5. The emergent behaviour when all agents move in the same directions seems to be even more competitive than with the tangential evasion heuristic. Agents tend to evade more often when the path is not free.

In dense crowds with multi-directional movement, pedestrians may use a different heuristic altogether. A commonly described phenomenon in these scenarios is lane formation, which has also been reproduced in simulation (e.g., Helbing and Molnár, 1995; Moussaïd et al., 2011). In these models, agents implicitly follow the paths used by others. I propose to make this behaviour explicit. If the path to the target is not free some steps ahead, agents try to follow a peer walking in the same direction. If the path to this leader is also blocked or no other agent in the proximity walking in the same direction can be found, the sideways evasion heuristic is used. The follower heuristic is shown in figure 6.6.

For these heuristics, some cognitive capacities are necessary. Individuals have to be able to know where they are in the environment (e.g., McNaughton et al., 2006), anticipate the next step (e.g., Schiff and Oldak, 1990), and detect collisions (e.g., Schiff and Detwiler, 1979). For the tangential and sideways evasion heuristic, individuals need the additional capacity of estimating distances (e.g., Ziemer et al., 2009 and Iosa et al., 2012). Finally, in the follower heuristic, the capacity of assessing motion is used (e.g., Warren and Rushton, 2009 and Fajen et al., 2013). All of these capacities

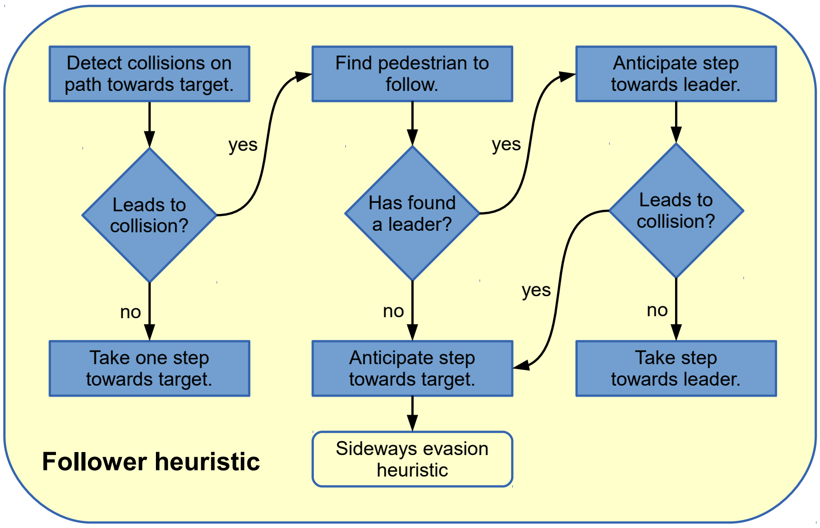

Figure 6.6: Flow chart for the follower heuristic. If a collision on the path to the target is detected, the focal agent tries to find the next agent moving in the same direction. If such a leader has been found, the next step is taken in the direction of them. If no leader can be found or the step towards the leader also leads to a collision, the agent resorts to the sideways evasion heuristic. (Figure: Seitz et al., 2015a)

can be assumed in humans and many other animals. For example, Baba et al. (2010) observed a behaviour similar to the step or wait heuristic in invertebrates.

For the implementation in a computer simulation, an algorithm that captures the behaviour has to be defined. The heuristics I proposed mainly rely on geometrical computations and binary decisions. For the tangential evasion heuristic, this is shown in algorithm 1. The algorithms for the other heuristics follow a similar structure of checking alternatives if the preferred position leads to a collision. In contrast to models based on systems of ordinary differential equations such as force-based models heuristic decision making is based on simple calculations and distinction of cases. Neither a numerical solver for ordinary differential equations nor an optimisation scheme is employed.

The heuristics are not very demanding computationally and do not employ numerical optimisation. No analogies such as social forces or utility optimisation are used. The computational steps are simple, and heuristic building blocks can be combined to form new, more complex decision processes. For example, the tangential evasion heuristic uses the step or wait heuristic if evasion is not possible. The sideways evasion heuristic extends the tangential evasion heuristic with additional movement options, and the follower heuristic uses the sideways evasion heuristic as a fallback if following a leader is not possible. Therefore, I argue that the model follows the paradigm of cognitive heuristics as proposed by Gigerenzer and Todd (1999). Before I discuss simulation results in subsection 6.3.3, I give some details on the software implementation.

Algorithm 1 Algorithm for the tangential evasion heuristic, a model of pedestrian decision making. The algorithm is invoked for each individual pedestrian movement step in the discrete stepping process or for each numerical step in the force-based continuous process on the physical layer (chapter 5). The result in the form of the next preferred position is stored in x_p.

```
determine target direction
compute next position x_target towards the target
if x_target leads to a collision then
    store agent A the collision occurs with
    compute both tangential evasion steps x_a and x_b in relation to A
    choose the evasion position x_a with smaller distance to the target
    if x_a leads to a collision then
        if x_b leads to a collision then
            remain at the current position: x_p = x_current
        else
            go to the second evasion position: x_p = x_b
        end if
    else
        go to the first evasion position: x_p = x_a
    end if
else
    take step towards target: x_p = x_target
end if
```

6.3.2 Implementation details

I refer to the software implementation of the cognitive heuristics model as *behavioural heuristics model* (BHM). The overall class structure is integrated into the framework Vadere the same ways as the optimal steps model (section 4.4, chapter 4). The general class diagram is shown in figure 6.7. The class `PedestrianBHM` is derived from `Pedestrian` and reused for all heuristics. It provides the basic functionality for the stepwise motion with the event-driven update scheme. Specific heuristics are implemented through the interface `Navigation`. The interfaces `DirectionAddend` and `SpeedAdjuster` are necessary for compromise decision making.

Figure 6.8 shows realisations of the interface `Navigation` that implement the cognitive heuristics model described in the previous subsection. The step or wait heuristic and the tangential and sideways evasion heuristics are implemented in the class `NavigationPorximity` together because they share the same basic algorithm. `NavigationCluster` is another heuristic still in development. All navigation schemes use computations implemented in static methods of the class `UtilsBHM`.

Figure 6.9 shows the class diagram for the interfaces `DirectionAddend` and `SpeedAdjuster`. Although both are not necessary for the model described in the previous section, they provide the necessary structure for extensions to the decision-making model. They allow for compromise decisions by adjusting the speed and direction. The model of small-group coherence I describe in section 7.2 of the next chapter makes use of these interfaces. The three interfaces specific to the behavioural heuristics model

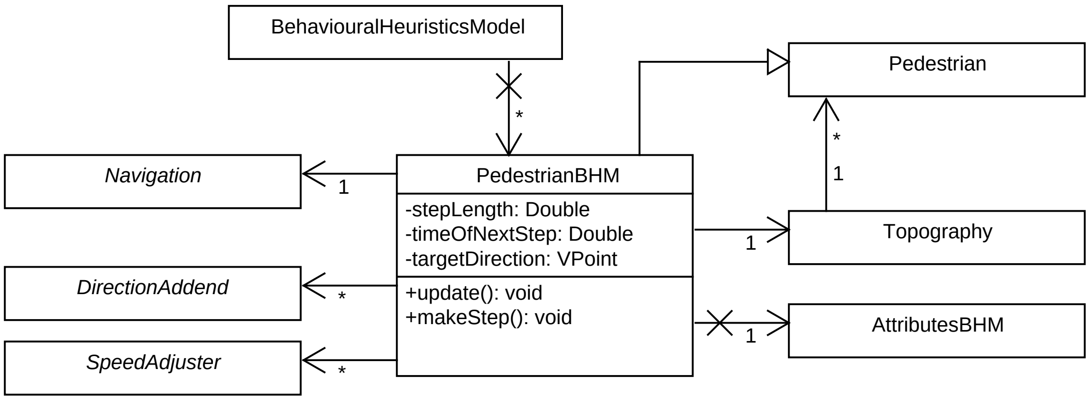

Figure 6.7: Class diagram of the behavioural heuristics implementation. The simulation loop triggers the update of all agents by calling the update function of the object of type `BehaviouralHeuristicsModel`. The class `PedestrianBHM` has access to the position of other agents through the topography object. Model parameters are set in the class `AttributesBHM`. Different heuristics models can be defined using the interface `Navigation`. The interfaces `DirectionAddend` and `SpeedAdjuster` provide functionality for additional compromise decisions in direction and speed, which are not part of the simulation model yet. These kind of decisions are necessary for small-group behaviour (section 7.2, chapter 7) or keeping space between agents and obstacles.

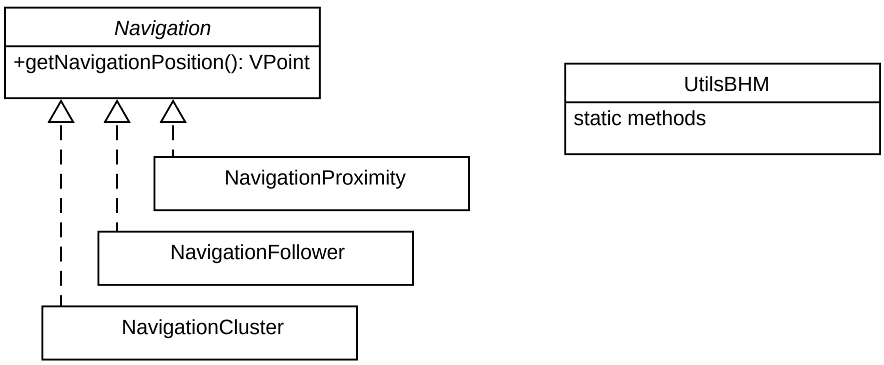

Figure 6.8: Class diagram of the interface `Navigation` and the class `UtilsBHM`. The class `NavigationProximity` contains the routines for the step or wait heuristic and the tangential and sideways evasion heuristics. The cluster heuristic is a possible extension to the model I present in this work. The static methods in `UtilsBHM` are used by the navigation classes.

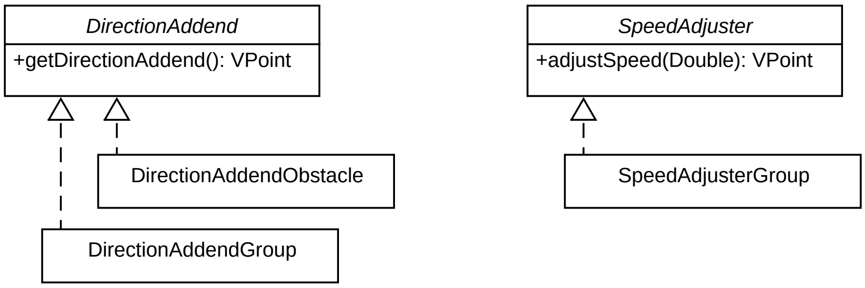

Figure 6.9: Class diagram of `DirectionAddend` and `SpeedAdjuster`. Both implementations are necessary for compromise decisions such as small-group coherence or distances kept between agents and obstacles. The speed adjuster gets the current speed and increases or decreases it according rules defined in the model. The direction addend changes the current direction by adding a vector to it.

allow for fast model extensions with new heuristics and compartmentalise the software according to the heuristic building blocks. The static methods in `UtilsBHM` can be reused by all classes.

6.3.3 Simulation results

We simulated and analysed two scenarios to study the model based on cognitive heuristics (Seitz et al., 2015a). The first is an egress scenario where agents have to pass through a bottleneck and a congestion forms in front of the entrance to it. The same scenario was studied in a controlled experiment before (Liddle et al., 2009) for which the data describing the trajectories of participants is available online (University of Wuppertal, 2015). This scenario is fundamental in safety studies because bottlenecks are a possible hazard in an evacuation. In the second scenario, agents are placed in a corridor walking in opposite direction. Therefore, they encounter oncoming agents who they have to evade. This scenario is also important for safety considerations because contra flow can lead to jamming, which may cause dangerous crowd densities or even crushes. A phenomenon that is usually observed in this scenario is the formation of lanes.

For the bottleneck scenario, we used two measures to quantitatively describe the result. One is the density-speed relation measured at a rectangle with 1 m side lengths directly in front of the bottleneck. We determined the density using Voronoi cells (Steffen and Seyfried, 2010) and the speed with a running average (Seitz and Köster, 2014). We developed a second measure M_q to describe the formation in front of the bottleneck, which reveals whether agents form a disciplined queue or a less ordered congestion in the shape of a half circle. The quantity is

$$M_q = \sum_{i=1}^{n} \frac{1}{1+x_i} \times y_i = \sum_{i=1}^{n} \frac{y_i}{1+x_i}, \tag{6.1}$$

where y_i is the lateral distance to the centre of the entrance to the bottleneck and x_i the vertical distance (figure 6.10, left). The index i runs over all n agents who have

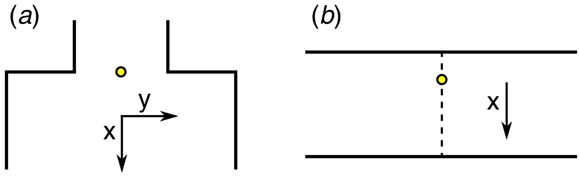

Figure 6.10: Schematic illustration of the two simulation scenarios with reference points in yellow. Part a of the figure shows the bottleneck scenario where agents are created on the bottom and pass through the corridor on the top. The distance to the reference point (in yellow) is measured in x and y-direction to obtain the quantity M_q. Part b shows the contra-flow scenario where agents moved both from right to left and left to tight. One possible reference point is highlighted (in yellow). In every time step, multiple reference points are chosen on the halfway mark (dashed line), and the quantity M_l is computed for all of them. The distances x_i in M_l are calculated with respect to the reference points.

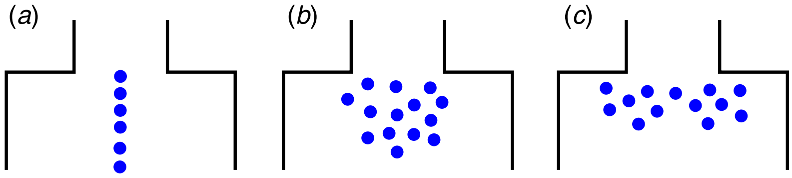

Figure 6.11: Illustration of different formations in front of a bottleneck. In part a, the queue measure M_q would yield almost 0 because pedestrians do not deviate from the centre of the corridor laterally. In part b, it would yield a higher value as pedestrians also walk to the side of the entrance. In part c, an even higher value is obtained for M_q.

not yet passed the entrance. Hence, when agents tend to walk to the sides of the entrance, the measure M_q becomes greater. Agents farther away from the entrance in the vertical direction are weighted less. For the shape of the congestion in front of the bottleneck, this means that greater values of M_q correspond to a wider congestion and smaller values to a more narrow formation. In the extreme, when all agents perfectly queue horizontally in the middle of the entrance, M_q would be 0 (figure 6.11).

We simulated a scenario with 180 agents who are created 8 m away from the bottleneck. The bottleneck has a width of 2 m and a length of 5 m. We compared the simulation outcome to the data from the controlled experiment (Liddle et al., 2009; University of Wuppertal, 2015). Figure 6.12 shows snapshots of the simulation and experiment in the first row, the queue measure M_q in the second row, and the density-speed relation in the third row. The simulation was run for the step or wait heuristic (a), the tangential evasion heuristic (b), and the sideways evasion heuristic (c). The follower heuristic yielded very similar behaviour to the sideways evasion heuristic and hence its results are not shown here.

The step or wait heuristic allows for egress without jamming, but it takes the longest until all agents have left the room. The tangential and sideways evasion heuristic lead to similar egress times that also match the experimental data. The queue measure in the experiment, however, shows greater similarity with the tangential evasion heuristics than with the sideways evasion heuristic. The step or wait heuristic

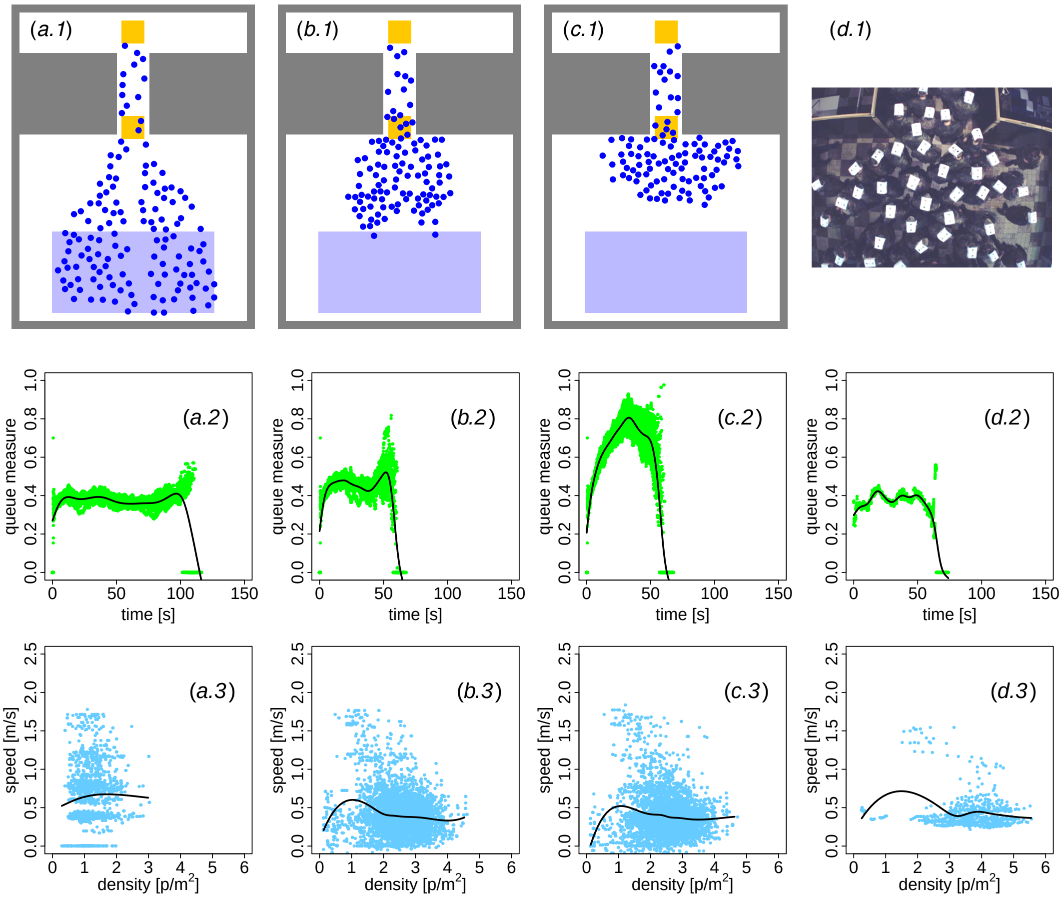

Figure 6.12: Simulation results for a bottleneck scenario compared to a controlled experiment with similar set-up. the first row shows snapshots of the simulation and experiment after 30 s. The second row reports the queue measure M_q described in the text. The third row shows the density-speed relation measured in a square rectangle directly in front of the bottleneck. The simulation was run for the step or wait heuristic (a), the tangential evasion heuristic (b), and the sideways evasion heuristic (c). The follower heuristic showed very similar behaviour to the sideways evasion heuristic and hence is not shown here. With increasing tendency to evade other agents in front to the side, the queue measure takes greater values. The experimental data is matched best with the tangential evasion heuristic. (Figure: Seitz et al., 2015a. Snapshot and data in panels d: University of Wuppertal, 2015)

produces the least competitive emergent behaviour because agents do not overtake. Therefore, the congestion in front of the bottleneck is narrow. With increasing tendency to overtake in the tangential and sideways evasion heuristic, the queue measure becomes greater and the congestion broader.

In different situations, pedestrians may behave differently and thus use different heuristics to make decisions. Specifically, the step or wait heuristic seems cooperative in the sense that agents do not try to overtake but wait when the path in front of them is blocked. In an emergency situation, individuals may become more competitive and, therefore, try to overtake others. This actually leads to faster egress in the simulation, although egress times do not improve for even higher competitiveness with the sideways evasion heuristic. Since the heuristics contain each other, the potential cognitive demand is increasing from a to d, that is, more cognitive effort seems necessary for more competitive behaviour.

The different formations in front of the bottleneck seem to reflect the degree of competitiveness. Similar shapes have been observed recently in an experiment where participants were instructed in a way that they behaved very competitively (Pastor et al., 2015). Interestingly, the degree of competitiveness and the shape of the congestion in the controlled experiment agrees with the interpretation of the tangential evasion heuristic being less competitive than the sideways evasion heuristic.

For the contra-flow scenario, we simulated a 48 m long and 6 m wide corridor. Agents are created at both ends of the corridor and try to reach a target placed at the other end. Agents are introduced into the corridor at different creation delays to obtain a variety of densities (one agent on both sides every 0.5, 1.0, or 1.5 seconds). Again, we used two measures to describe the simulation outcome. For the first quantity, we recorded the flow of agents at the halfway mark. This allowed us to compare the efficiency of the heuristics. The second quantity M_l measures the degree of lane formation in the corridor. The crossings of agents at the halfway mark are weighted, and contributions of one direction are assigned positive signs and the other direction negative signs. The individual contributions are weighted both in time and space:

$$M_l = \sum_{i=1}^{n} I_i \times \exp\left(-\frac{t_i}{10}\right) \times \exp\left(-x_i\right), \tag{6.2}$$

where I_i indicates the direction of agent i with 1 or -1, t_i the time distance of the crossing of agent i to the measurement point, and x_i the lateral distance to the measurement point when agent i crossed the halfway mark (figure 6.10, right). The quantity M_l is recorded and changes over time and lateral distance.

Figure 6.13 reports the simulation results for the step or wait heuristic (a), the tangential evasion heuristic (b), the sideways evasion heuristic (c), and the follower heuristic (d). The first row shows snapshots taken after 100 s with a delay of 0.5 s. The second row shows the flow rate at the halfway mark averaged over 10 simulation runs. The third row reports the queue measure for one simulation run with a delay of 1.0 s.

The step or wait heuristic leads to immediate jamming in the corridor when the agents walking in opposite direction meet because they do not evade to the side. This can be seen in the snapshot (*a.1*) as well as in the flow rate (*a.2*) and lane formation

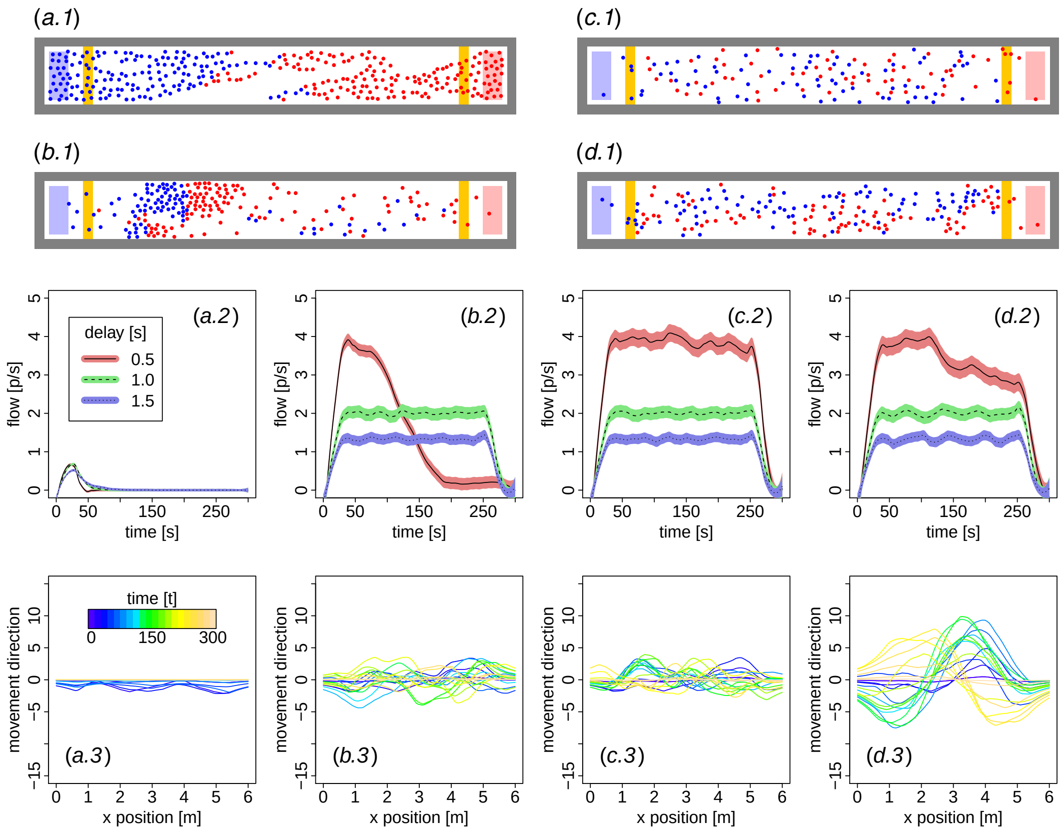

Figure 6.13: Simulation results for a corridor scenario with a length of 48 m and width of 6 m. Agents are introduced into the scenario at the ends of both sides and try to reach the target located on the other end of the corridor. One agent is created on each side with a delay of 0.5, 1.0, or 1.5 seconds. The first row shows snapshots at 100 s after the start of the simulation with a delay of 0.5 s. The second row shows the flow rate averaged over 10 simulation runs for each delay. The third row reports the lane formation measure M_l described in the text for one run with a delay of 1.0 s. The simulations were conducted with the step or wait heuristic (a), the tangential evasion heuristic (b), the sideways evasion heuristic (c), and the follower heuristic (d). The step or wait heuristic leads to an immediate jam when pedestrians walking in different directions meet because they do not evade but remain at the current position. Occasional jams occur with all heuristic at the highest density (delay of 0.5 s). However, the sideways evasion heuristic led to only one jam over 10 simulation runs. Although the follower heuristic is more likely to produce jams, it shows the highest degree of lane formation. (Figure: Seitz et al., 2015a)

measure (*a.3*). The tangential evasion heuristic and follower heuristic also lead to occasional jams with the highest rate (delay of 0.5 s). Only the sideways evasion heuristic allows for flow without jams – except for one case with a delay of 0.5 s where a jam occurred, too. Although with the follower heuristic jams are more likely than with the sideways evasion heuristic, it produces a higher degree of lane formation, a phenomenon that can be observed in real crowds (Kretz et al., 2006a; Moussaïd et al., 2012).

The step or wait heuristic does not seem to describe a plausible behaviour in this scenario. Nevertheless, it produced plausible results for the bottleneck scenario, which suggests that different pedestrian behaviours apply in different situations. This conclusion is also supported by the follower heuristic and the sideways evasion heuristic. They did not show much difference in the bottleneck scenario, but in the corridor, the follower heuristic is the only one that produces a considerable degree of lane formation.

In terms of cognitive effort, the follower heuristic is potentially more demanding than the other three. However, following a leader in front may also reduce cognitive demand when successful. In that case, following another person in front may, in fact, be less demanding than the two evasion heuristics. The simplest behaviour, the step or wait heuristic, does not allow for continuing flow in the corridor and hence must be dismissed for this scenario. This suggests that cognitive demand in contra-flow scenarios is higher for the least demanding behaviour that still allows ongoing flow.

Table 6.1 summarises the heuristics with the emergent behaviours in the two scenarios, the potential cognitive effort, and the cognitive capacities necessary for their computation. The results demonstrate that simple heuristics that do not rely on numerical optimisation can produce emergent phenomena of pedestrian dynamics. The heuristics describe behaviour more directly and thus are accessible to researchers not proficient in formal sciences. This is an important aspect because pedestrian dynamics is a very interdisciplinary field. Cognitive heuristics allow for the study of hypotheses beyond mere pedestrian dynamics, such as considerations of cognitive demand and effort. This shows how representing the underlying processes is beneficial for scientific studies in new directions. Since we demonstrated that the paradigm of cognitive heuristics can be used to reproduce known phenomena, facilitates extensions and new developments based on findings in other domains, can be tested well in experiments, and generates new hypotheses both in pedestrian dynamics and beyond it, I consider the approach suitable for the simulation of pedestrian dynamics. In the next subsection, I discuss remaining challenges, extensions, and future directions.

Heuristic	**Definition**	**Emergent behaviour in Bottleneck scenario**	**Emergent behaviour in Contra-Flow scenario**	**Potential cognitive effort (ordinal scale)**	**Cognitive demand**
Step or wait heuristic	Pedestrians anticipate the next step but only take it if it does not lead to a collision.	Pedestrians do not overtake or walk around others, passive queueing.	Immediate congestion when pedestrians walking in opposite direction meet.	1	Anticipate step towards target, detect collisions
Tangential evasion heuristic	If the next step leads to a collision, pedestrians try to avoid it tangentially.	Pedestrians sometimes try to overtake and walk around others, no queueing.	Congestion with higher densities, minor lane formations	2 (contains step or wait heuristic)	+ determine tangential evasion directions, estimate distances
Sideways evasion heuristic	If tangential evasion fails, pedestrians then try to avoid the collision to the sides.	Pedestrians very frequently overtake and walk around others, no queueing.	Least likelihood of congestions, least lane formations	3 (contains tangential evasion heuristic)	+ determine sideways evasion directions
Follower heuristic	If a collision on the path towards the target is detected, pedestrians follow another individual walking in the same direction.	Similar to the chosen proximity evasion heuristic, active queueing if no proximity evasion is used.	Moderate likelihood of congestion with high densities, strongest lane formations	4 (contains sideways evasion heuristic)	+ determine walking directions of other pedestrians, select another pedestrian to follow

Table 6.1: Summary and comparison of heuristics for pedestrian behaviour. (Table: Seitz et al., 2015a)

6.3.4 Future directions

The simulation approach with heuristics has proven to be suitable for the simulation of pedestrian dynamics. The paradigm is backed by research in psychology and provides vast flexibility for new directions. However, some issues exist that I discuss in this subsection and for which I suggest solutions.

Different heuristics may apply in different situations. Therefore, the question arises how the correct heuristic is selected. Gigerenzer (2008) discussed this question and suggested that the most likely way is through reinforcement learning. Perhaps the mind can be seen as one great combination of heuristics and selecting an appropriate one depending on the context is a heuristic that uses other heuristics as building blocks too.

In model development, this flexibility makes it hard to falsify the paradigm, because new heuristics can easily be introduced to extend existing ones. Therefore, every concrete heuristic should be seen has a theory, and once it has been shown to be wrong, it either must be dismissed or changed. The paradigm of heuristics has to be studied on another level, such as through experiments in psychology and neurology or by evolutionary consideration. Furthermore, cognitive heuristics have already been studied thoroughly, which makes them a reliable theory that can be built on.

How can we study whether a model of decision-making actually describes a real process? So far, we have defined a model, simulated emergent behaviour, and compared it to controlled experiments. Additional studies of the same kind should be conducted to further evaluate the proposed heuristics. This way, whether the respective heuristics are used can only be deduced from the emergent phenomena. There are two general approaches for collecting evidence of a specific heuristics being used as cognitive strategy:

> Two main research approaches have been developed to investigate and identify people's strategies. One, the process-oriented approach, focuses on the predecisional process by looking, for example, at the order of information acquisition; the other, the outcome-oriented approach, focuses on the outcomes of the decision process and builds models that predict these outcomes. (Rieskamp and Hoffrage, 1999, p. 142)

Information retrieval can be captured in situations where people have to actively obtain the information or by monitoring their eye movement (Rieskamp and Hoffrage, 1999). The former seems less suitable for microscopic pedestrian decisions but could be applied for route choice tasks where information is distributed in the environment and people have to actively move around in order to obtain it. Another possibility could be controlled virtual environments (Bode et al., 2014; Bode and Codling, 2013) where every decision is recorded, which may reveal the information search process. Monitoring eye movement seems to be technically challenging for pedestrians walking in real environments but is already feasible for immersive virtual environments (Steptoe et al., 2008).

In general, verbal descriptions could be given during the decision process or after it. While both options have the limitation that subjects might not be able to give accurate accounts of their own decision process, the latter seems to be even more susceptible to obfuscation (Rieskamp and Hoffrage, 1999). The method is problematic in general considering that decisions are not necessarily made consciously (Nisbett and Wilson, 1977) and, therefore, are not accessible for verbalisation. In fact, even our goals and motivations are often unconscious (Custers and Aarts, 2010). Intuitively, it seems that we do not know how we make movement decisions. Usually, we just make them. Decisions about route choice, on the other hand, could be seen as more rational and might be based on conscious decisions. Nevertheless, whether we think that we know how we decide is not a trustworthy indicator: experiments have shown that subjects tend to invent narratives that are evidently wrong whenever they do not know how they made the decision (Nisbett and Wilson, 1977).

For some emergent behaviours, another kind of computation may be necessary that has not been used for the heuristics I proposed in this chapter. For example, it can be observed that pedestrians keep a certain distance to obstacles and other pedestrians (e.g., Moussaïd et al., 2009b; Seitz et al., 2014b). The distance pedestrians keep to each other also correlates with the density and flow of pedestrian streams and thus is an important aspect of pedestrian dynamics. To reproduce this behaviour, agents have to compromise between keeping a distance to obstacles or other agents and moving forward fast. This tradeoff is easily observable at corners where pedestrians want to pass as fast as possible but also do not want to get too close to the wall (Seitz et al., 2015b). Hence, they stay away from the wall but get closer at the corner because this saves time.

Another example is the behaviour of small social groups. It has been shown that individuals who want to communicate with each other tend to walk in a specific formation (Moussaïd et al., 2010). By adjusting the speed and direction of agents, the formation can be reproduced implicitly in a simulation without explicitly imposing it (section 7.2 of the next chapter). To realise this behaviour with a heuristics model, a compromise decision between keeping the formation and moving forward is necessary. For instance, if the group formation does not fit into a corridor, it has to be given up in order to pass through.

6.4 Remaining, waiting, and queueing

Pedestrians are usually understood as moving and, therefore, measures such as the flow are used in pedestrian dynamics. However, in many scenarios, pedestrians remain at a position for a variety of reasons. For example, they may be waiting for the bus or a train. Queueing can be seen as a special form of waiting without necessarily remaining at one position for longer time intervals. Other forms of remaining may have ends in themselves, including stopping to chat with someone or dancing at an open air festival. I argue that the aspect of waiting and, more generally, remaining is an important niche in the study of pedestrian dynamics.

For an observation of shuttle buses, we statistically analysed the time it took passengers to board the bus (Torchiani et al., 2015). In the observation, we registered the time points whenever a passenger entered the bus. We analysed the data from a series of observations and found that there is a tendency of a group of passengers entering quickly and additional groups entering with some delay (figure 6.14). To

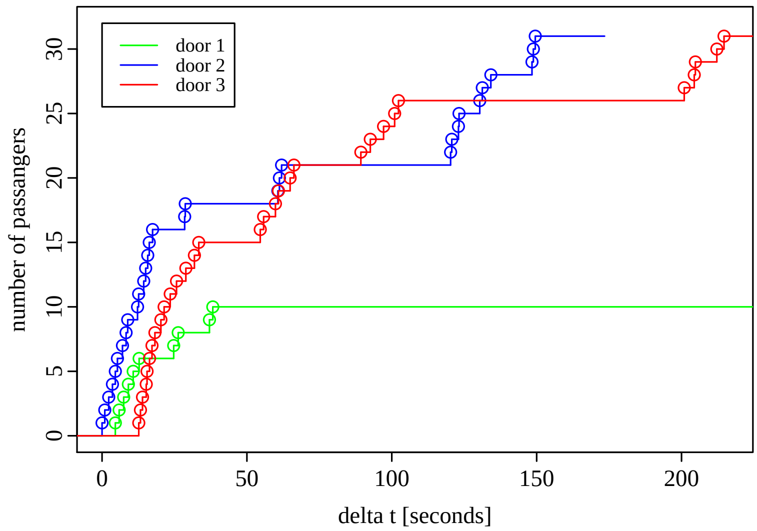

Figure 6.14: Descriptive statistical analysis of an observation of passengers entering a shuttle bus. Each circle represents one passenger entering the bus. At first, waiting passengers who queue at the door enter the bus quickly. Then, other groups enter the bus with delays. This finding can be used for the optimisation of public transportation systems. (Figure: Torchiani et al., 2015)

describe this phenomenon, we proposed two linear regressions, one for the first group and one for the following ones that arrive after some time threshold. This finding has an impact on the capacity utilisation because not all passengers enter immediately after the bus has arrived but some rather with delay. In our report, we also proposed ideas on how this kind of data can be used in microscopic simulation together with a graph-based network optimisation for scheduling the bus arrivals and departures.

Waiting behaviour seems particularly relevant in transportation systems. Knowledge about how passengers wait can help to improve capacity utilisation, comfort, and even safety. Therefore, we dedicated a conference contribution[2] to the analysis of waiting behaviour at a railway station platform. Figure 6.15 shows an impression of the platform at which the data was collected. For our analysis, we defined "*waiting* as a type of behavior by individuals remaining at a position to pass time until an event they expect occurs" (Seitz et al., 2015c). This definition excludes queueing and other forms of remaining at one position, including any leisure activities.

At first, we drew on the literature in social sciences as background for the study of waiting behaviour. For example, we referred to Ruesch and Kees (1956) and Rapoport (1977), who studied the meaning of space and the impact the built environment can have on our behaviour. Hall (1966) proposed characteristic social distances we keep depending on our relation to the other person – namely, the intimate, personal, social, and public distance around us at up to 0.45, 1.2, 3.5, and 7.6 metres, respectively.

After reviewing the background from social sciences, we analysed data collected at a railway station platform to complement the considerations. The observation area was a railway station platform in Vienna. The data had been collected by annotating

[2]The remaining section is based on this collaborative work (Seitz et al., 2015c).

Figure 6.15: Impression of the railway station platform. The photo was taken from a different position than the position of the camera which recorded the data. (Figure: courtesy of Stefan Seer)

the positions of passengers in the video footage that was captured from an oblique location above. Thus, the coordinates had to be transformed. A transformed snapshot of the camera view is shown in figure 6.16. The platform had been observed twice, once in the morning at 7:00 am (38 passengers) and once in the evening 6:30 pm (91 passengers). The platform is 7 m wide.

To summarise which areas were used for waiting, we counted the number of passengers in cells with 1 m side lengths every second, summed the count up, and divided the sum by the overall observation time in seconds. This gave us a measure of occupancy shown in figure 6.17 (left panels). The colour map reveals that passengers did not wait close to the escalators on the left nor did they get too close to the platform edge. In addition to the location, we were interested in the time passengers remained at one position, that is, the remain time. We averaged the remain time over the cells, which is shown in figure 6.17 (right panels). We did not find a systematic distribution of remain times.

In a second step, we examined the data in more detail with a series of histograms shown in figure 6.18. The first row reports the positions chosen by passengers measured from the lower platform edge. Again, it can be seen that passengers stayed clear of the platform edge, and mostly maintained a minimum distance of 1 m. The second row shows the distribution of remain times of all passengers, which resembles an exponential distribution. The minimal distances to the next passengers are reported in the third row. In the evening, the distances seem to be lower, which could be because more people knew each other than in the morning. However, it could also simply be an effect of the higher crowd density in the evening. The last row shows the absolute distance to the closest platform edge, which supports the finding that passengers mostly kept a minimum distance of 1 m.

Waiting pedestrians have largely been ignored in pedestrian dynamics and especially in simulation models. Johansson et al. (2015) presented a mechanism for the social force model that makes agents wait at a specific position. However, they do not

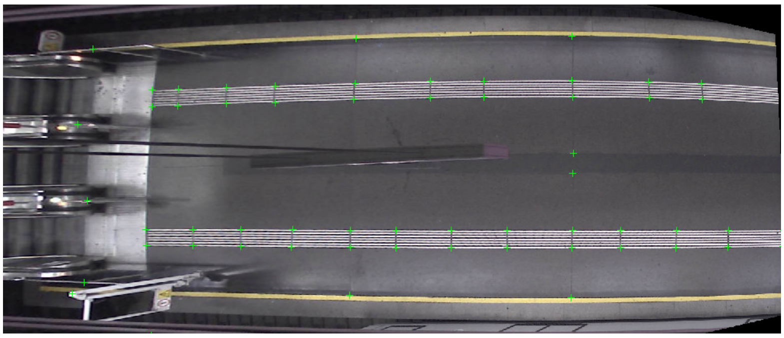

Figure 6.16: Transformed view of the camera that was used to record the data we analysed. The escalators can be seen on the left; the two parallel, white horizontal lines are the tactile pavement; the yellow lines indicate a safety distance to the train. (Figure: courtesy of Stefan Seer)

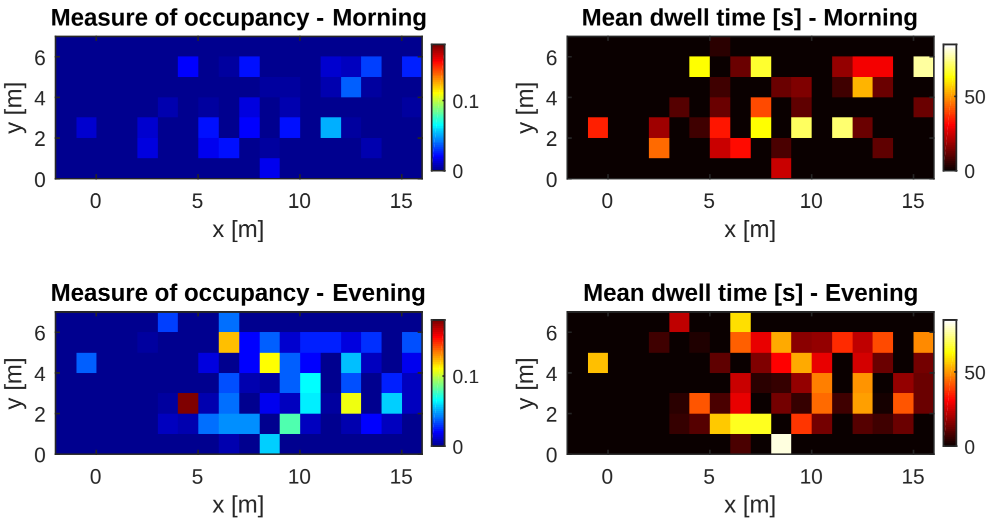

Figure 6.17: Colour maps of the railway station platform in cells with 1 m side lengths. On the left, we summed up the number of passengers in one cell every second and divided it by the overall observation time. On the right, we averaged the remain time of individual passengers over the cells. In general, passengers stayed clear of the area next to the escalators on the left and the platform edge on the top and bottom of the colour maps. No systematic distribution can be observed for the remain times. (Figure: Seitz et al. 2015c)

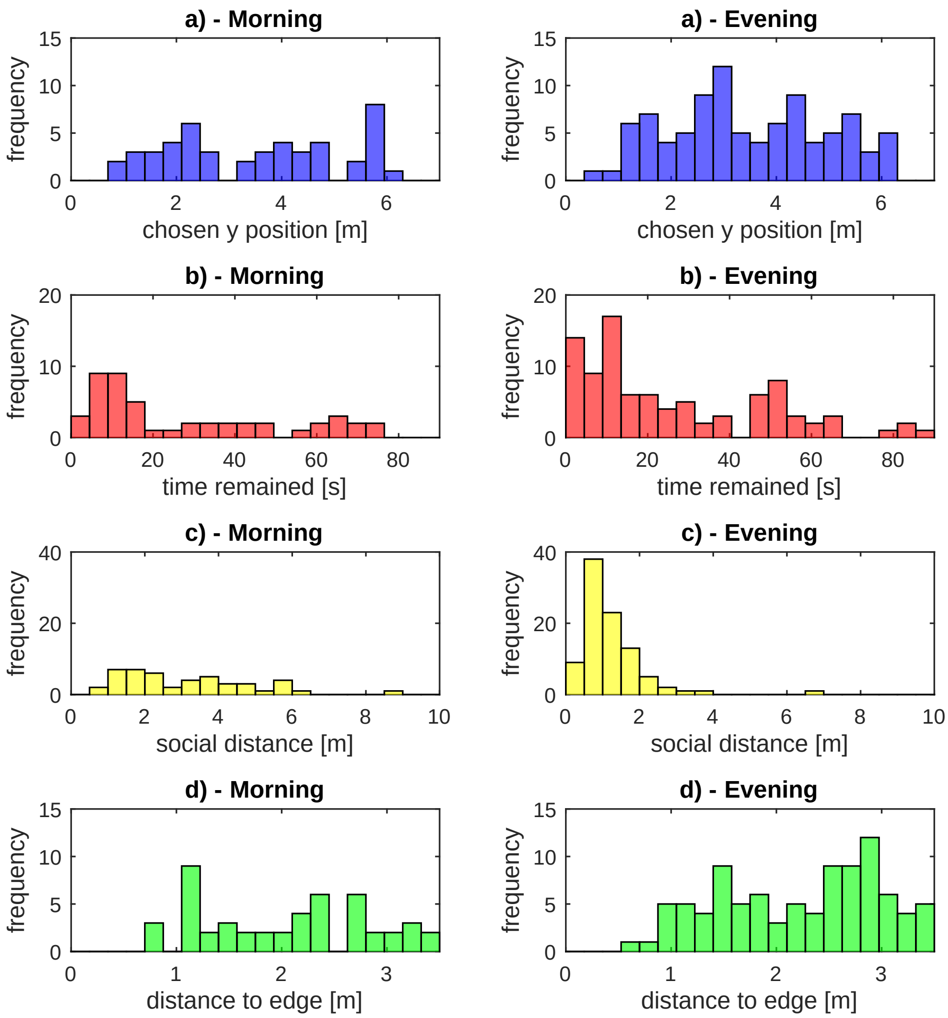

Figure 6.18: Histograms describing the behaviour of passengers at a railway station platform in the morning (left column) and the evening (right column). The first row shows the distance they kept from the lower platform edge, and the last row shows the distance they kept to the closest edge. The second row reports the times passengers remained at one position. The third row shows the distances to the next waiting passengers at the position they remained. These statistics support the finding that passengers stayed away from the platform edge. The distribution of remain times resembles an exponential distribution. In the evening, passenger appeared to keep less distance, which could be because of more social groups, such as friends, but may also simply be a result of higher densities. The data, particularly the distances and remain times can be used in models of waiting behaviour. (Figure: Seitz et al. 2015c)

provide a model of how agents choose this position. Davidich et al. (2013) studied a simulation scenario at a large railway station. They introduce polygonal waiting zones in a cellular automaton, which have to be explicitly defined and arranged for every simulation scenario. Pedestrians who reach the waiting zone and wait there choose a random position within the waiting zone.

In public transportation systems, waiting behaviour may have a great impact on the overall performance, capacity utilisation, and comfort of passengers. Thus, it seems especially important that waiting behaviour be included in simulations studying these scenarios. The findings from the analysis we conducted may be used to develop a model for the simulation of pedestrian behaviour. The measures can be employed for the validation and comparison of the simulation outcome with empirical data. The findings themselves may serve for the optimisation of existing systems, and the methodology can be applied for future studies of waiting behaviour in different scenarios.

To go in the direction of simulation models, we proposed four heuristics that may describe a more general behaviour of passengers at railway station platforms:

> a) passengers get close to where the train arrives; b) they keep a safety distance to the platform edge; c) passengers keep a social distance to other passengers; d) they stay away from the escalators. (Seitz et al., 2015c)

Although this is still a rather informal description of behaviour, specific parameters could be set given our quantitative analysis or by measuring them explicitly at the studied site. Once a formal model has been developed, it can be used for the simulation of waiting behaviour employing existing locomotion and proximity navigation models, including the optimal steps model (chapter 4) and the behavioural heuristics model (subsection 6.3.1, chapter 6). While still at an early stage, the introduction of waiting agents in pedestrian stream simulations seems an important niche. With our empirical studies, we laid the necessary basis for simulation models of waiting pedestrians.

6.5 Summary

I covered the psychological layer of microscopic crowd simulation in this chapter. This layer explicitly does *not* represent the locomotion process but the decision-making aspects of pedestrian behaviour. Every pedestrian simulation has a representation of the decision-making process although it is often not made explicit. The separation of locomotion and decision making allows for more flexibility and detail in the respective domains.

In section 6.1, I drew on the literature that provides insights into individual behaviour, namely psychology and animal behaviour. I discussed important general concepts of psychology, including perception, decision making, and behaviour. There is a wealth of research on all of these topics, and especially perception could be an interesting direction of future research. Psychology is a broad field and relevant aspects have to be identified for a pedestrian stream simulation. In animal behaviour, two research areas seem to be particularly relevant for pedestrian dynamics. The first is collective behaviour and self-organisation. Here, also simulation models of swarms, flocks, and schools have been proposed. The second is animal navigation, although it may be concerned more with route choice than proximity navigation.

I discussed the concepts of bounded rationality and cognitive heuristics in subsection 6.1.3. Although there is some debate on how heuristics are used and whether they work well or not, the fundamental paradigm is widely accepted. The fast and frugal heuristics school has come up with a specification of what a cognitive model constitutes. Specifically, they dismiss numerical optimisation as a plausible concept for human decision making. Because of its wide acceptance and aptitude to develop a flexible decision-making model, I chose the paradigm of cognitive heuristics for the development of the decision-making layer.

The behavioural sciences aim to describe and predict natural behaviour. In artificial intelligence (section 6.2), the objective is to engineer a programme that shows intelligent behaviour. This area can also be of interest for the development of pedestrian simulations. Especially the agent concept seems to provide a good start for individual-based simulation approaches. Some other fields in artificial intelligence may be interesting for the study of pedestrian dynamics, such as planning, animation, and robotics.

In section 6.3, I discussed decision-making models for the simulation of pedestrian dynamics. The optimal steps model (chapter 4) uses local utility optimisation, a concept that may not agree with modern psychology. However, it can still be a useful model because of its conciseness, efficiency, and validity in emergent phenomena. Following the paradigm of bounded rationality, I proposed four heuristics that describe individual behaviour in pedestrian crowds (subsection 6.3.1). The heuristics use computational steps that can be assumed as human cognitive capacities. They are modular and may be used as building blocks in other heuristics. This allows for the flexible development of new models but also makes it easy to test parts without necessarily changing the rest of the model. The implementation of the behavioural heuristics model in the simulation framework Vadere is modular, too, which facilitates code reuse and extension (subsection 6.3.2).

The simulation results of a bottleneck scenario and a contra-flow scenario (subsection 6.3.3) provide evidence that the paradigm is suitable for simulation studies in pedestrian dynamics. Different heuristics seem to apply in different contexts, which led to hypotheses about the cognitive demand associated with different scenarios. More competitive behaviour seems to entail potentially greater cognitive effort. The step or wait heuristic, which has minimal cognitive effort, was dismissed as a plausible model of contra flow in a corridor because it led to immediate jams even for very low densities.

The possible combinations of locomotion and decision making models I developed are summarised in table 6.2. The optimal steps model corresponds to combination *a* and the simulation approach for the results shown in subsection 6.3.3 to *b*. I used combination *d* for the simulations in subsection 5.3.2. The group-level models in the next chapter can be combined with different locomotion and decision-making models.

The discussion raised some issues that I covered in subsection 6.3.4. An open question remained how heuristics are selected depending on the context, which is an interesting direction of research. Another issue was that it may seem as if the

		Decision making	
		Utility optimisation	Cognitive heuristics
Locomotion	Discrete steps	a	b
	Force-based	c	d

Table 6.2: Possible combinations of locomotion and decision-making models for the respective two layers. Combination a corresponds to the optimal steps model, b to the simulation approach used in subsection 6.3.3, d to the simulations in subsection 5.3.2. The decision-making models presented in this chapter are highlighted in yellow.

model could not be falsified because of its flexibility. It is true that the paradigm of heuristic decision making must be studied on another level, but the heuristics proposed can be tested very well as has been shown in the simulation study. I gave an outlook on how the question whether these particular heuristics not only reproduce emergent effects but also are actually used by humans can be studied. In the future, compromise decisions may be necessary to simulate some phenomena, including small group coherence. This aspect is discussed again in the next chapter on social and group behaviours.

In pedestrian dynamics, mostly the motion of crowds is studied. In reality, pedestrians often remain at a position. Especially waiting and queueing is common in public transportation systems, which I discussed in section 6.4. Remaining for leisure activities such as dancing or chatting can be observed at mass events. In an observation of a shuttle bus station, we analysed how long it took passengers to get on the bus. Using data from another observation of a railway station platform, we studied the waiting behaviour of passengers. We proposed informal heuristics that describe the behaviour of pedestrians in the observation. The measurement methodology and findings can be used to develop a simulation model.

Chapter 7

The social layer: Collective behaviour

The social layer covers various aspects that build on the psychological and physical layer (figure 7.1). The psychological and physical layer have to provide the basic functionality so that agents can navigate in their proximity of the environment. The separation of the social and psychological layer may not be necessary for all studies. In some sense, the behaviours described in the previous section are social interactions too and cannot be clearly separated from the models in this chapter. Nevertheless, the separation remains helpful for the modularity of the overall concept.

For the scientific background of this chapter, I mainly draw on social psychology and studies in collective behaviour from biology, which sometimes deal with the same topics. It is a great challenge to formalise concepts from social psychology for computer simulation, especially, since social behaviour heavily depends on the context. Models from computational biology can be useful as they are already formalised. However, the disparity among species must not be neglected: humans live in a complex social environment that is not matched by any other animal.

The studies in this chapter cover aspects of social behaviour that can be seen as extensions to existing simulation approaches. This modularity has the advantage that the aspects can be investigated separately, and they may be integrated into one

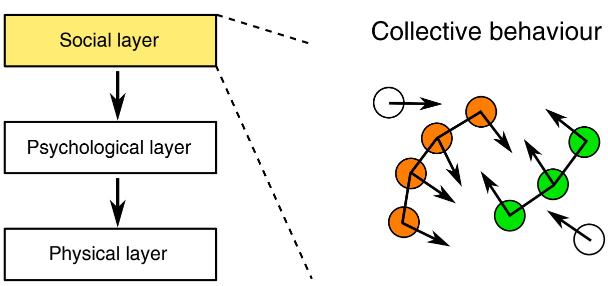

Figure 7.1: Illustration of the model separation into three layers. This chapter focuses on social aspects and builds on an existing psychological and physical layer.

coherent simulation approach afterwards. For example, in our work on how people search for others in a building (von Sivers et al., 2015), the locomotion and proximity navigation model is the optimal steps model. We complemented it with a graph-based navigation layer that lets agents search through the building. Different strategies can be used for the search: the optimal route, a random route, or a route generated by a heuristic. The heuristic selects the closest room to the current position that has not been visited yet. We argued that this is the most plausible model of human decision making compared to the other strategies. Real human behaviour has many facets, but this model is a first step in the direction of a full simulation describing how humans search a building.

In the first section of this chapter, I review aspects from social psychology and collective behaviour. The review gives some general background from which the social identity model is revisited in section 7.3. In section 7.2, I cover sub-group behaviour, meaning the spatial formation small groups of up to four pedestrians exhibit within a crowd. The sub-group behaviour influences the overall pedestrian stream (Moussaïd et al., 2010; Köster et al., 2011b) and hence is an important extension to individual interactions. In addition, I show how the somewhat looser coherence of larger groups can be reproduced. In section 7.3, I discuss how concepts of crowd psychology could be integrated into a computer simulation and what theoretical challenges this poses.

Simulating social and collective behaviour is especially challenging. The complexity of human behaviour (section 1.3 in the introduction) becomes even more pronounced in social psychology. Here, behaviour strongly depends on the context and thus is difficult to predict. Furthermore, to simulate collective behaviour of pedestrians, a reliable locomotion and decision-making model has to be available. The models of small-group coherence are concrete and have been used in simulations of pedestrian dynamics. The results discussed in the study on how to introduce models from social psychology are still preliminary.

7.1 Aspects from social psychology and collective behaviour

Social psychology is the "branch of psychology that studies the effect of social variables on individual behavior, attitudes, perceptions, and motives; also studies group and intergroup phenomena" (Gerrig and Zimbardo, 2002). Therefore, it is highly relevant for pedestrian dynamics. In biology, collective behaviour is mostly studied based on the concept of self-organisation. "The central tenet of self-organization is that simple repeated interactions between individuals can produce complex adaptive patterns at the level of the group" (Sumpter, 2006). However, the view of biology on collective behaviour as being the result of simple interactions has sparked criticism from social psychology:

> The crowd in whatever guise is often an object of controversy in the wider society. Consequently, there is a clear ideological function now, as there was a century ago, in attributing to the crowd some essential primitive quality explained in terms of the conceptual frameworks of the biological

> sciences rather than the social sciences, which sets the crowd apart from other forms of social life. Rendering the crowd a primitive, biological object, an elemental force, deprives the behaviour of its members of meaning, agency, and legitimacy. In academia, psychology's inferiority complex in relation to its 'hard science' neighbours, and the rise of cognitive science, neuroscience, evolutionary psychology and complexity science, have kept alive the fantasy of a unifying 'life science', whereby the behaviour of human crowds and all other collective phenomena—from bee swarms to social innovations—can be adequately captured by a single set of biologically grounded simple rules. (Drury and Stott, 2011, p. 276)

To better understand this criticism: a general distinction between physical and psychological crowds can be made. Physical crowds are a collective of individuals who do not necessarily have a common goal or intention to cooperate but are a "mere aggregate of individuals or small groups" (Drury et al., 2009). The behaviours simulated in the previous chapter emerge out of merely individual goals. Cooperation was only an emergent result, that is, individuals did not explicitly cooperate. The members of a psychological crowd, on the other hand, do have a conception of the group they are in on some level (e.g., Drury et al., 2009). Discussing collective behaviour, Searle (2002) argues that "collective intentionality presupposes a background sense of the other as a candidate for cooperative agency, i.e. it presupposes a sense of others as more than mere conscious agents, but as actual or potential members of a cooperative activity." The distinction is important in crowd psychology because in this field, a *crowd* is usually understood as a psychological crowd and not a mere aggregate of individuals.

Human behaviour depends on the context: whether we go for a walk as a leisure activity or find ourselves in an emergency evacuation makes a difference. Especially the public character of crowds – whether physical or psychological – is an important factor. Matsumoto (2012) found that "being in the public eye" has the greatest impact on our behaviour. Group behaviour is determined by the context, too. A group that is not confronted by outsiders may not even recognise itself as a group (Turner et al., 1994). When confronted with outsiders, the group identity can become salient. The concrete behaviours may then be determined by social norms, that is, the "expectation a group has for its members regarding acceptable and appropriate attitudes and behaviors" (Gerrig and Zimbardo, 2002).

One of the first authors who popularised crowd psychology was Le Bon, who wrote in his book *La Psychologie des Foules* from 1895:

> What constitutes a crowd from the psychological point of view—A numerically strong agglomeration of individuals does not suffice to form a crowd—Special characteristics of psychological crowds—The turning in a fixed direction of the ideas and sentiments of individuals composing such a crowd, and the disappearance of their personality—The crowd is always dominated by considerations of which it is unconscious—The disappearance of brain activity and the predominance of medullar activity—The lowering of the intelligence and the complete transformation of the sentiments—The transformed sentiments may be better or worse than

> those of the individuals of which the crowd is composed—A crowd is as easily heroic as criminal. (Le Bon, 1996)

This excerpt illustrates some important points. Le Bon described members of a crowd as not having a personality or even "brain activity", which seems intuitively a distorted image of crowds and highly biased towards a negative perspective on collective action. Consequently, this view has been criticised as depicting the crowd as inherently pathological (Stott and Reicher, 1998). Le Bon's ideas, although never substantiated, can still be found in discussions on crowd behaviour today.

In contrast to the notion of "panic behaviour" (e.g., Helbing et al., 2000a), modern crowd psychology suggests that individuals in crowds can show high solidarity and cooperative behaviour (Drury et al., 2009). They assert that "mass emergency behaviour is both social (rather than antisocial or asocial) and cognitive (i.e., based on reasonable beliefs rather than non-cognitive emotions or instincts)" (Drury and Stott, 2011, p. 283). An explanation for this behaviour can be found in the social identity approach described in the following paragraph. Drury and Stott (2011) give a concise overview of crowd science in the context of emergency behaviour.

The social identity approach (e.g., Reicher, 1996; Drury and Reicher, 1999, 2010) presumes multiple social identities in addition to the individual identity we have. The social identities describe the group memberships such as profession, nationality, and so on. However, they are not all salient at the same time but rather *can* become salient depending on the context. For example, when confronted with out-group members, a sense of the group membership may become stronger. In order for the group to act as a collective, some requirements are necessary, such has group members knowing who belongs to the group and what behaviour is endorsed by the group. The process of how social identities become salient is described by self-categorisation theory (Turner, 1982). We categorise ourselves into social groups depending on the context, and thereby, the social identity of the respective group becomes salient.

7.2 Sub-group coherence

The approaches I discussed in the previous chapter on the psychological layer simulate individual behaviour neglecting any social structure of the crowd. A series of empirical studies has shown that crowds usually consist of sub-groups with up to four members (James, 1953; Coleman and James, 1961; Aveni, 1977; Singh et al., 2009; Moussaïd et al., 2010). Larger groups may also be present in the crowd but perhaps are more difficult to identify because of their loose coherence. Moussaïd et al. (2010) demonstrated that the formation of small groups is caused by the desire of members to communicate with each other. It seems likely that larger groups cannot maintain such a formation and hence decompose into smaller units with loose coherence (Moussaïd et al., 2010; Reuter et al., 2014).

7.2.1 Models of sub-group behaviour

Some approaches for sub-groups in microscopic simulations of pedestrian dynamics have been proposed. Singh et al. (2009) used attractor points around the group's centre

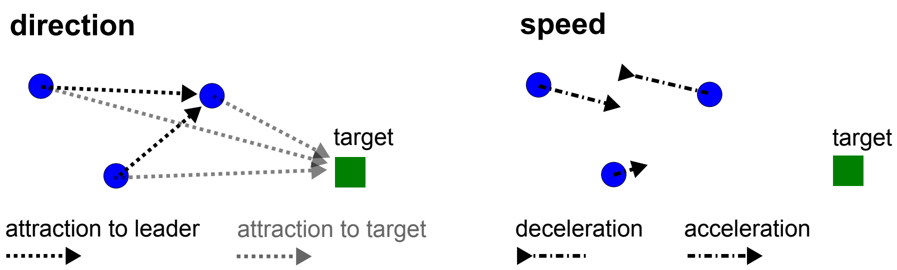

Figure 7.2: Illustration of the mechanisms in the sub-group model for groups of up to four members. The group formation is maintained through the manipulation of direction and speed. The direction is adjusted by agents following the most advanced group member while they also still move towards the target (left part of the figure). The leader is not one fixed agent but instead changes if another member becomes the one closest to the target. In addition, the speed is adjusted so that members falling behind slightly speed up and members ahead slow down (right part of the figure). (Figure: Seitz et al., 2014a)

to make members stay in the group formation. Qiu and Hu (2010) employed a matrix that describes the group relations among members. This flexible structure allows for different group behaviours, including linear and leader-follower formations. Moussaïd et al. (2010) used the social force model and introduce an additional acceleration term that makes agents adjust their position. The formation itself is not modelled explicitly but implicitly through the desire of individuals to communicate with others. The communication is the more comfortable the less pedestrians have to turn their heads. Therefore, agents try to minimise the head turn angle. With this model, the authors obtained the empirically observed formations. Zanlungo et al. (2014) proposed a potential that produces similar shapes. In animal behaviour, Langrock et al. (2014) studied the group formations of reindeer and proposed a model to describe their movement behaviour. Apart from scientific studies, group formations are also a topic of artificial intelligence, especially animation and gaming. Pottinger (1999a,b) described the model used in the real-time strategy game Age of Empires.

Inspired by the model of Moussaïd et al. (2010), we proposed an approach that we implemented for cellular automata (Köster et al., 2011b; Seitz et al., 2014a). We also assumed that group members want to communicate with each other. Instead of an attraction to the centre of the group, we let agents falling behind follow the group member most advanced towards the target (the leader). To restore the group formation, members ahead slow down and members behind slightly speed up. The attraction to the leader is realised by an additional attractive scalar field around the leader. The mechanisms for this approach are illustrated in figure 7.2. This model represents a simplification compared to the original model of Moussaïd et al. (2010) but at the same time reproduces the formations to a certain degree. In terms of cognition, this may even be a more plausible model as the behavioural rules are simple and describe behaviour explicitly. The model can be summarised with two rules: speed up if fallen behind and slow down when leading ahead of the centre of the group (first rule); follow the most advanced member of the group (second rule).

The agents simulated with this behaviour show visually realistic behaviour. They

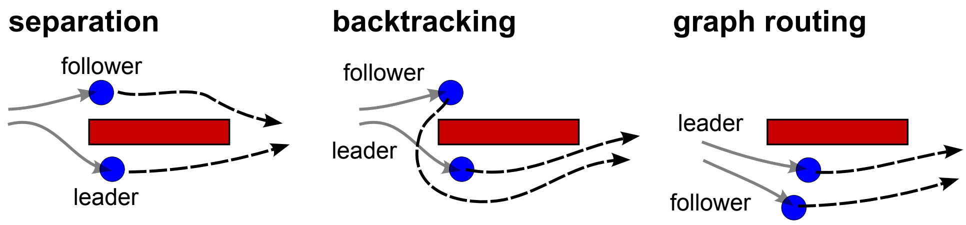

Figure 7.3: Illustration of the three resolutions when a group member is lost. Separation lets agents act as if the lost member were not part of the group anymore, which is computationally the most convenient solution that prevents unrealistic stalling of separated groups. Backtracking lets lost members go back to the position where they saw the leader last and try to rejoin the group from there. With graph-bases routeing, the whole group chooses the same path to navigate around the obstacle, which prevents separations. The last option seems to be the most likely behaviour in real pedestrian dynamics. (Figure: Seitz et al., 2014a)

stay together and form empirically observed formations when the path is free. The formation is not imposed explicitly but is an emergent effect induced by the behavioural rules encoded in the manipulation of direction and speed. When in a dense crowd or faced with a narrowing, the agents are able to give up the group formation to navigate more efficiently.

When the group is separated by an obstacle, the behaviour becomes unrealistic if the group members try to maintain the formation because they are still attracted to the leader. Since the attraction is calculated by using the Euclidean distance, they try to approach the leader through the target, which they cannot. Leading agents around the obstacle directly to the leader does not seem realistic either because it is implausible to assume that they know where the leader is exactly. Consequently, the leader and other group members ahead will wait for the lost member, and the whole group stalls. In this case, some mechanism has to adjust the behaviour to either separate the lost member from the group, let lost members track back to find the leader again, or prevent separation by assigning one path around the obstacle to the whole group (Seitz et al., 2011).

The first option, separating lost members, is the simplest solution and prevents unrealistic stalling of separated groups. Backtracking requires members to keep track of the position where they saw the leader last and setting the target to this position if the leader is lost. Assigning one path to the whole group can be realised with graph-based routeing (e.g., Kneidl, 2013). All of the options describe behaviour that may occur in reality. It seems most likely that group members, whether explicitly or implicitly, communicate the way around the obstacle among each other and then choose the same path. Figure 7.3 illustrates the three options.

In addition to the coherence of groups of up to four members, we proposed a model of larger groups (Seitz et al., 2014a). The larger groups comprise small groups and only the leaders of the larger groups are attracted to the most advanced member. Figure 7.4 shows simulation results using this approach. Larger groups in the simulation led to more jamming in a cross road simulation. We also studied the behaviour of groups

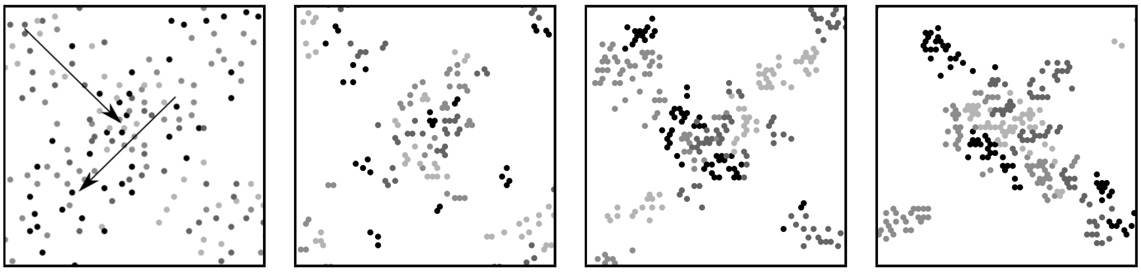

Figure 7.4: Simulation of agents in a cellular automaton walking from the top left and right corners to the corners diagonally across at the bottom. Group sizes are (from left to right): individuals, two to four members, groups of 14 members, and groups of 24 members. The groups are indicated with different shades. For the simulations, the model of small and large groups was used to achieve group coherence. Members of a group stay together, and with increasing group sizes, the congestion in the middle becomes more pronounces. (Figure: Seitz et al., 2014a)

comparing the simulation with empirical findings together with sociologists (Reuter et al., 2014). We proposed validation tests and scenarios for sub-group behaviour as a step in the direction of benchmark tests for microscopic pedestrian stream simulations (Köster et al., 2014).

The number and size of sub-groups vary across scenarios. For example, at a railway station at peak hour, probably fewer groups of families and friends are present than at a football match. Therefore, the distribution is a scenario specific parameter and ideally measured for the scenario before simulation studies are conducted. When the parameter is known, the models I discussed can be used for simulation studies. Sub-groups are an important extension of microscopic pedestrian stream simulation after it has been demonstrated that real crowds do not consist only of individuals. Furthermore, sub-groups are a first step in the direction of a more realistic representation of crowds with social relations among its members instead of a mere aggregate of individuals.

7.2.2 Implementation details

Originally, I implemented the approach described so far in this section for a cellular automaton that employed a scalar field. Since the optimal steps model also builds on the concept of scalar fields, the group model can easily be adapted for it. Figures 7.5 and 7.6 show the implementation in the software framework Vadere. The interfaces `GroupModel` and `Group` allow for the use of different group models. The model I presented above realises the interfaces in the classes `CentroidGroupModel` and `CentroidGroup`. The latter class also determines the group leader after each time step. Implementations of the group model interface create groups but are not a pedestrian factory. Instead, they are notified whenever a new pedestrian is created.

In the case of the optimal steps model, the group coherence is achieved by manipulating the scalar field (or "potential field") and the speed. Every object of type `PedestrianOSM` has an object of type `PotentialFieldPedestrian` describing the scalar field for the influence of other pedestrians and an object of type `SpeedAdjuster`.

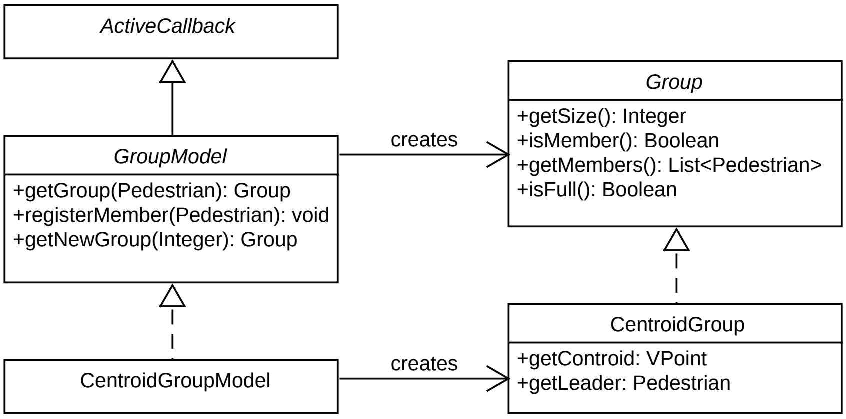

Figure 7.5: Group model class diagram. The interface `GroupModel` extends the interface `ActiveCallback`. Objects of type `GroupModel` register as listeners to the topography and are notified whenever a new Pedestrian is created. The interface definitions allow for the generic use of different group models. The object of type `GroupModel` creates objects of type `Group`. The model I presented in this section is implemented in the classes `CentroidGroupModel` and `CentroidGroup`.

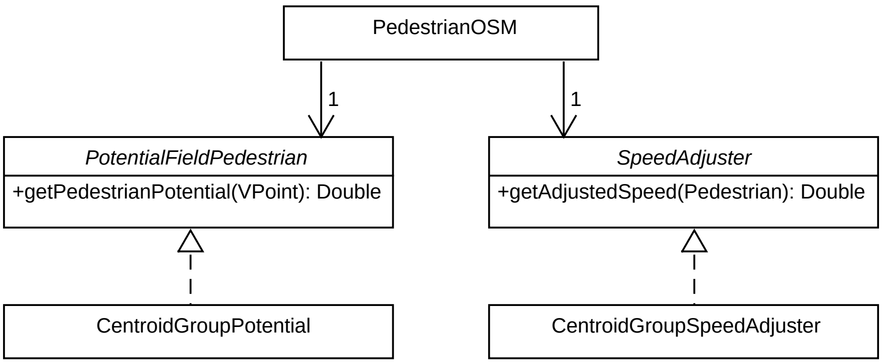

Figure 7.6: Class diagram for groups in the optimal steps model. Objects of type `PedestrianOSM` use one object of each type `PotentialFieldPedestrian`, which provides the scalar field taking into account other agents, and `SpeedAdjuster`. The group coherence is achieved through the manipulation of the scalar field and the speed. This is realised through the two group model implementations `CentroidGroupPotential` and `ControidGroupSpeedAdjuster`. They use the information stored in the objects of type `Group`, which store the group memberships of individual agents.

The group model manipulates both the scalar field and the speed through these objects to achieve the group coherence. For this, the classes use the information provided by the objects of type `Group`, which store the group memberships of individual agents. The influence of the group leader is introduced by the addition of an attractive function to the scalar field implemented in `PotentialFieldPedestrian`, which usually only has repulsive contributions from other agents.

In the optimal steps model, the group formations tend to be more precise than in cellular automata because of the movement in continuous space. For the behavioural heuristics model proposed in the last chapter, the adaptation seems more difficult at first as the model does not use scalar fields. However, the introduction of compromise decisions can be used to realise it. The vector pointing in the direction of the next step has to be combined with the vector pointing to the leader. The speed can be adjusted the same way as in the optimal steps model since both can use an event-driven update scheme with discrete steps (section 5.3, chapter 5). I describe the software structure for the behavioural heuristics model with compromise decisions in chapter 6, subsection 6.3.2.

7.3 Towards the integration of crowd psychology into computer simulation

So far, I have proposed models of locomotion and individual navigation in the proximity. Based on the models for individuals, I discussed mechanisms for the coherence of small groups. The notion of humans following simple rules has been criticised in social psychology (section 7.1). Social psychologists argue that human behaviour is driven by processes that neither ignore the individual as acting being nor the group level structures as fact. The following quote characterises this issue:

> For an adequate formulation of the individual-group relation, we need a way of describing group action that neither reduces the individual to a mere target of group forces of mystical origin, nor obliterates the organized character of group forces in the welter of individual activities. We need a way of understanding group processes that retains the prime reality of individual *and* group, the two permanent poles of all social processes. We need to see group forces arising out of the actions of individuals and individuals whose actions are a function of the group forces that they themselves (or others) have brought into existence. We must see group phenomena as both *the product and condition* of actions of individuals. We cannot resolve the difficulty by merging the two extreme views in some judicious way. To overcome the apparent contradictions it is necessary to take into account what both positions lack: an understanding of the fact of psychological interaction and the mutually shared field that it produces. (Asch, 1952, pp. 250–251)

In this section[1], I discuss the interface between computer simulation and social sciences in the context of crowd behaviour, especially crowd psychology. Sime (1995) pointed out the importance of social psychology for engineering over 20 years ago. Since then, many simulation approaches have been proposed that are also used for the simulation of evacuation scenarios. However, most of them still focus on individuals and not so much on the group level (Templeton et al., 2015). Perhaps this is the case because there are still great challenges in developing a reliable pedestrian simulation on the individual level so that valid group-level behaviour, which requires a basic locomotion and navigation layer, was too difficult to implement.

Nevertheless, some authors have taken steps in this direction. Aguirre et al. (2011), proposed to use agent-based models (ABM) as basis: "Contemporary ABM computer programming emphasising the characteristics of individuals and their propensities to act should be supplemented with codes informed by group level dimensions such as norms, values, commitments to lines of action, leadership, and a sense of identification and membership in meaningful groups." Using the optimal steps model, von Sivers et al. (2014) simulated an emergency scenario with agents helping others, that is, walking back to them and finally egressing together. Later, they used the simulation to study the impact of the tendency to help on evacuation times with uncertainty quantification (von Sivers et al., 2016). Uncertainty quantification renders it possible to systematically study the distribution of the outcome given a probability distribution of the input parameters (e.g., Smith, 2014).

In general, simulation approaches seem to lack a group level representation that is also recognised by agents (Seitz et al., 2015d). The sub-group models discussed in section 7.2 are a first step in that direction. However, they reproduce only local emergent behaviour and are inapt for generic collective mechanisms. For example, two groups of football fans that both share a social identity within their respective groups and act accordingly cannot be captured with the previous models. An emergent behaviour of the social identity one group shares could be that they accept smaller distances to in-group members (Novelli et al., 2010).

I argued that there is no theoretical contradiction that makes introducing concepts from psychology to computational sciences impossible (section 1.3 in the introduction). While simulation models have to be defined in a strictly formal way with mathematical equations or algorithms, models and theories in psychology may be less formal. Nevertheless, both fields rely on testing theories and validating models; both aim at describing the same world we live in. The challenge lies on a more practical level: how can models from psychology be formalised and captured in an algorithm? This may not be possible practically when the model is too extensive or unclear, but it should be possible in principle unless the theories are contradictory.

A possible avenue for future research could be the separation in layers that helps to focus on the respective fields of expertise. On the psychological layer, the paradigm of bounded rationality and cognitive heuristics may facilitate the formalisation of models from social sciences as it provides a basis that agrees with a modern understanding of human cognition. More specific requirements for the integration of group processes in

[1] Some of the ideas and the line of argument in this section were developed in collaborative work with researchers at the University of Sussex (Seitz et al., 2015d).

crowd simulations could be a mental representation of the group, its members, norms, and other features in every agent. I end this section with a quote from a philosophical perspective on collective behaviour that may also inspire future work in mathematical model development:

> Ask yourself what you must take for granted in order that you can ever have or act on collective intentions. What you must suppose is that the others are agents like yourself, and that they have a similar awareness of you as an agent like themselves, and that these awareness coalesce into a sense of *us* as possible or actual collective agents. And these conditions hold even for total strangers. (Searle, 2002)

7.4 Summary

This final main chapter dealt with the social layer. The social layer builds on an existing locomotion and decision-making model. The psychological layer could not be clearly separated from it because every pedestrian interaction has social aspects. Nevertheless, the distinction seemed useful to modularise modelling approaches and separate aspects of behaviour that build on existing models of pedestrian navigation such as the optimal steps model.

I discussed background from social psychology and collective behaviour in section 7.1. How collective behaviour is demarcated from other social interactions could not be clearly defined. In biology, any emergent group phenomenon based on local rules seems to be considered collective behaviour. In social psychology, some authors specifically criticised the view that collective behaviour arises out of mere simple and local interactions. Additionally, historic work on crowd psychology such as that of Le Bon, who claimed humans in a crowd show a "disappearance of brain activity", has been rejected as out-dated. A theory in social psychology that provides a broader view of collective behaviour is the social identity approach.

In section 7.2, I presented models of sub-group behaviour, that is, groups of up to four members. Empirical research has shown that people in a crowd often move in groups. The group size distribution depends on the scenario. The simulation approaches build on an existing locomotion and decision-making model for pedestrians and complement them with mechanisms that lead to agents within the group staying together. Originally, our models were developed for a cellular automaton but can also be integrated with the optimal steps and behavioural heuristics model. I outlined the software implementation in our framework.

In section 7.3, I discussed how crowd psychology can be introduced in computer simulations. Specifically, the criticism from some social psychologist that computer simulations are reductionist was considered. Finally, I suggested that the separation into layers and the paradigm of cognitive heuristics may facilitate the introduction of concepts from social psychology.

Chapter 8

Summary, conclusions, and future directions

This final chapter is divided into three sections. In section 8.1, I summarise the work presented in the previous chapters and review what has been accomplished. In section 8.2, I critically discuss the work, especially revisit the challenges of capturing human behaviour (section 1.3 in the introduction). Finally, I address open questions and give an outlook on possible future directions in section 8.3.

8.1 Summary

The aim of this work was to investigate and develop pedestrian stream simulations that capture both emergent phenomena and the underlying processes. I implemented the models in software, ran simulations, compared the outcome to empirical data, and investigated the models conceptionally to assess their suitability for the representation of the natural processes. For model development, I drew on the literature from the respective domains, including biomechanics, cognitive sciences, and social psychology. Where necessary, I initiated controlled experiments or used available empirical data for the calibration and validation of the models.

In the introduction (chapter 1), I gave general background on computational science and engineering with a focus on model development. Although it is not always clear how to compare competing models, their usefulness for a specific purpose can indicate how to choose a model. My work focused on computational methods for pedestrian dynamics with a special focus on the underlying natural processes. A series of phenomena has been reported in pedestrian dynamics both from simulation studies and empirical observations. Most important measures are the density-speed relation, the degree of lane formation in contra-flow scenarios, and egress times. There are challenges when trying to model human behaviour because it is complex and can often only be described stochastically. Furthermore, the methodology and terminology vary across disciplines. However, no fundamental contradiction could be identified that would prevent using models from psychology for simulating pedestrian behaviour. For this work, I chose a separation of modelling aspects in a physical, psychological, and social layer. The separation is clearly an idealisation of the real world but is justified and helpful for model development and implementation.

In chapter 2, I presented the software framework Vadere, which I developed together with colleagues. Our objective with the framework was to provide a research platform that allows for fast model changes and new developments. We chose an agile development process and a clear separation of the pedestrian models from other elements in the simulation. This way, it is possible to maintain an arbitrary number of simulation models without interference of one another. The simulator was complemented with a graphical user interface that allows for parameter specification, file management, graphical scenario creation, and visualisation of the simulation outcome. The software architecture has proven to be suitable for the fast changing requirements in a research project through a series of model developments, including the ones I presented in this work.

The next chapter (chapter 3) was dedicated to a review of existing approaches for the simulation of pedestrian dynamics. I only considered microscopic models since macroscopic representations of pedestrian behaviour did not seem detailed enough to capture the underlying processes. I classified the modelling concepts in cellular automata, velocity-based models, force-based models, and other approaches. Cellular automata are highly efficient in computation but do not provide a high-resolution account of pedestrian dynamics. Especially the cell structure leads to artefacts that have to be dealt with. Velocity-based and force-based models represent motion continuously and determine behaviour through a system of ordinary differential equations. Both are mathematically concise, which also limits their capacity to capture a variety of behaviours with the same set of parameters. Alternative approaches stem from artificial intelligence and animal behaviour. I carried out an assessment of a class of models with the perspective of scalar fields. This allowed identifying the superposition principle as a common feature of a majority of models. However, I also pointed out conceptual limitations of this approach.

In chapter 4, I presented the optimal steps model. It is based on the stepwise movement of pedestrians, which is used for a natural discretisation of locomotion and decision-making. The step length of real pedestrians is also the motion step length of agents in the simulation. Through controlled experiments, we obtained the step length in relation to the speed of movement and used it as a parameter in the simulation. The possible steps in all directions from the current position define a circle around the agent. The next position on this circle is chosen by optimising a utility function. This approach is in line with the superposition principle of scalar fields. For the numerical solution, an equidistant grid was employed because of its robustness and efficiency in computation. In controlled experiments, we observed a constraint in the change of direction depending on the speed. We used the empirical upper bound in the change of direction as a constraint in the step circle. With an event-driven update scheme, the individual steps of agents can be ordered naturally. The implementation of the optimal steps model in Vadere allows for the study of different optimisation and update schemes. A series of validation experiments demonstrated the suitability of the optimal steps model for simulation studies. The limitations of cellular automata because of the cell structure are overcome while computational efficiency is maintained. Moreover, various extensions and developments showed that the model is both flexible and able to capture features of real pedestrian behaviour.

The first chapter (chapter 5) of part II was dedicated to the physical layer. I reviewed some background from biomechanics and robotics. Both disciplines are heavily researched, and hence, I cannot claim having identified every aspect that might be of relevance. I discussed some interesting concepts, such as dynamic walking, forward and inverse models, and body planes. For pedestrian dynamics, the transverse plane is used while in biomechanics mostly models in the sagittal or coronal plane are studied. Models in biomechanics are usually still too complex for pedestrian dynamics where often a large number of pedestrians is simulated. I presented two approaches for the simulation of pedestrian dynamics on the physical layer. The first is the discrete, step-wise motion scheme from the optimal steps model. The second is a force-based model that explicitly does *not* represent decision making but locomotion. The decision of where to step next has to be provided and, based on it, the force vectors accelerating the agents are constructed. This model allows for the introduction of other physical forces, such as contact forces, and yields a continuous motion process. Both models can be used on the physical layer in combination with different decision-making schemes. Specifically, the discrete stepping model can be combined with other decision-making models than utility optimisation, and the force-based model can be combined with both utility optimisation or heuristic decision making.

In chapter 6, I drew on the background from psychology, animal behaviour, cognitive sciences, and artificial intelligence to model the decision making of pedestrians. Again, all of these fields are extensive and I could not review them exhaustively. Key aspects had to be selected to keep the resulting models parsimonious. Psychology and animal behaviour provide findings for important aspects such as perception and emergency behaviour. In cognitive sciences, the idea of utility optimisation and other models of *un*bounded rationality have been criticised as an implausible representation of human decision making. Therefore, I followed the direction of *bounded* rationality and cognitive heuristics as a modelling paradigm. Here, decisions are made with simple rules that employ evolved biological capacities. Consequently, I proposed a model that captures decision making with heuristics and argued for each computational step that humans are capable of it. In two simulation studies, we validated the model and argued that it matches the requirements to be considered a plausible representation of human decision making, which was not the case for any other model discussed before. Waiting behaviour is an important aspect of pedestrian dynamics in many situations. We observed this behaviour in different scenarios, particularly the waiting behaviour of passengers on a railway station platform. Informal heuristic rules described the findings. The behaviour of waiting persons has been largely neglected in pedestrian stream simulation but is important, particularly for transportation systems.

In the last chapter (chapter 7), I discussed models of group behaviour that build on an existing physical and psychological layer. The scientific background was found in social psychology and collective behaviour. Social psychology deals with many aspects of human behaviour that are not all relevant for pedestrian dynamics. There is some debate about what collective behaviour is. Biologists tend to see it as self-organisation that emerges from local interactions while some psychologists argue that this view is reductionist and neglects the reality of groups. It is well-documented that crowds consist of sub-groups of up to four members in many scenarios. I presented a model for the coherence of sub-groups, described the implementation, and reported simulation results. I finally studied crowd models from social psychology and identified challenges in using them for pedestrian dynamics. The core issue does not seem to be a fundamental contradiction but the challenge of formalising models from social psychology.

The whole work represents an advancement towards a natural representation of pedestrian behaviour on different layers. The necessary tools and the modelling basis were laid out in part I. Specifically, an effective software framework and the knowledge of existing approaches is necessary to make progress in this discipline. The optimal steps model represents an improvement of cellular automata and introduced stepping behaviour as discretisation. I dedicated one section to each of the three layers and proposed different models that aim at capturing the underlying processes in part II. The following list summarises the models I presented for the respective layers:

Models on the physical layer:

- discrete stepping process
- continuous force-based process

Models on the psychological layer:

- utility optimisation
- heuristic decision making
- remaining, waiting, and queuing (conceptual)

Models on the social layer:

- sub-group coherence
- social identity approach (conceptual)

Every layer can now be studied independently by experts in the respective domains, and resulting models can be fit into the overall simulation. In the next section, I point out the achievements and remaining challenges and assess the results.

8.2 Conclusions

I implemented the models proposed in chapters 4 to 7 in the software framework Vadere. We developed the framework specifically for the purpose of research and teaching. The requirements were that it be flexible and provide as many reusable parts as possible. The implementation of the models in this work and the use of the framework for seminars and bachelor, master, and doctoral theses demonstrates the success of the software design (section 2.6).

The perspective of scalar fields helped to classify simulation approaches and identify similarities and differences (section 3.5). Although the superposition principle seems to have limitations, it has allowed for the reproduction of emergent crowd phenomena. The optimal steps model also follows the superposition principle but goes beyond previous approaches. It is discrete in time and space but not bound to a cellular grid. It is computationally efficient and exploits the natural process of stepping. This allows for the study of pedestrian stepping behaviour, which had not been introduced to simulations for pedestrian dynamics before. Although utility optimisation was criticised as an implausible representation of human decision making, it is still an accessible and concise concept. For the prediction of pedestrian flows without the need to understand the underlying process, the optimal steps model is an effective simulation approach.

I proposed two models for the physical layer: the discrete stepping process of the optimal steps model and a force-based process. Both can be combined with different decision-making concepts. For computationally efficient simulation, the discrete process is advantageous. If continuous motion or the combination with contact forces is desirable, the force-based process can be used. On the psychological layer, cognitive heuristics are a plausible concept for human decision making. We successfully used it to reproduce crowd phenomena in simulation studies, and it allowed us to develop novel hypotheses on cognitive effort. An open question that remained was how heuristics for specific behaviours are selected based on the context. A possible solution could be another heuristic that also decides on which behaviour to follow based on cues from the environment. Waiting and other forms of remaining are still a neglected niche in pedestrian dynamics. I argued that it should not be neglected in simulation studies both in safety engineering as well as in transportation science as pedestrians often remain at a position a great portion of the time. We proposed preliminary rules that describe the waiting behaviour of passengers at a railway station platform.

On the social layer, small-group coherence is an important extension to individual behaviour because it is frequently observed in real crowds. In addition, it may be necessary to introduce models from social psychology in order to capture advanced aspects of collective behaviour since there are some phenomena that cannot be explained well otherwise. With a theoretical treatment of the topic, we laid the foundations for this direction. Although challenges remain, the collaboration with social psychologists seems to be promising as the principle objective of understanding and predicting crowd behaviour is the same in both disciplines.

Simulation results have to be treated with care because individual human behaviour often cannot be predicted precisely. Our understanding of human decision making is still at an early stage in computational science and engineering, and hence, the models I proposed are a venture into largely unknown territory. However, though challenging, this also makes it interesting. For the use of the simulations, it is important to notice that hazards and problems can be found with simulation studies, but it is not possible to prove that no other unknown ones exist. This said, the application of the optimal steps model by the start-up company accu:rate has demonstrated its usefulness in a series of studies on safety and efficiency issues.

8.3 Future directions

The software framework Vadere is constantly being extended and refined. The basic architecture, however, remains unchanged. Extensions of the software framework could be a representation of additional scenario elements, including stairs, obstacles of different heights, agents with additional attributes, or cars and other vehicles. Some of these elements have already been tested in student projects. For a better exchange within the scientific community, we plan on publishing the framework as open source project.

The optimal steps model has already been extended in various projects (section 4.6). Studies that investigate the stepping behaviour of pedestrians are especially promising as this represents a unique feature of the model. For example, walking on stairs can be captured well since it clearly depends on the stepping behaviour. On the implementation side, some improvements are possible. For instance, the parallel algorithm we proposed could be realised for computation in a graphics processing unit on a video card. Alternative data structures and algorithms may further improve the computational performance.

The force-based model on the physical layer can be extended by contact forces to capture phenomena such as crowd turbulences. This would be important for the study of crowds with high densities. The physical representation of agents in the environment could be refined with ovals or more complex shapes. Models from biomechanics may be used to represent the locomotion process in greater detail. Also models in the sagittal or coronal plane are of interest for this as they capture specific features of pedestrian movement. For example, the swaying of heads is important for the analysis of video footage because, often, the heads of pedestrians are tracked, which leads to a sinusoidal trajectory.

To consolidate the modelling approach with cognitive heuristics, additional validation studies are necessary both with more empirical data for similar scenarios but also other scenarios. Not all aspects of pedestrian motion have been captured with the proposed heuristics. Examples of other behaviours are the evasion of whole groups of people, which would allow for a more forward-looking behaviour of agents. This points into the direction of graph-based routeing. The combination of heuristic decision making and a graph representation of possible paths could be an interesting development. For instance, in our study on how people search a building, we already proposed heuristic decision making as one solution. Another important issue is the selection of a heuristic based on the context. With additional background from literature in biology and psychology, such as on perception, new model aspects and refinements may be developed.

The hypotheses of necessary cognitive effort by pedestrians could be tested in other fields. Findings on cognitive effort and demand may help to improve the built environment – make it safer, more efficient, and comfortable. Whether specific heuristics are actually used by pedestrians can also be tested by other methods than video observation. For example, technologies for immersive virtual reality could help studying individual behaviour in fully controlled environments. This may also allow for the investigation of information-search strategies through eye tracking.

We proposed conceptual heuristics for waiting behaviours. Considering how important this behaviour is in real scenarios, the introduction of dedicated simulation models seems to be an interesting niche. The models could be combined with the optimal steps model, which would allow for general adaptation in many approaches based on scalar fields. Most interesting seems the formulation with cognitive heuristics as these directly state the decision rules and represent a plausible model of the underlying process. Additional empirical studies are necessary to deduce more general models that apply in a variety of scenarios. For example, cross-cultural studies may reveal differences in the waiting behaviour.

Some behaviours cannot be captured well without a group-level model. The social identity approach explains a variety of crowd behaviours. However, it still has to be formalised on a conceptual level to reproduce more than some specific behaviours predicted by it. Initial steps have been undertaken. Important next steps include the identification of requirements for an agent representation, the formalisation of the model, and the selection of suitable empirical data for validation. Cognitive heuristics could be one possibility to describe behaviours that result from a self-categorisation process in the social identity approach.

Finally, there are numerous fields that study further aspects that are relevant for pedestrian dynamics. Every direction provides findings that can be used for model development. One example are interdisciplinary studies in the direction of collective behaviour from a biological perspective. Promising applications are the study of transportation systems and urban spaces with the aid of pedestrian simulations.

Bibliography

Abbott, B. (1935). Herald Square, 34th and Broadway, Manhattan. Online: digitalcollections.nypl.org. From The New York Public Library (Reference : CNY# 145). Accessed: 8. January 2016.

Adenzato, M. and Garbarini, F. (2006). The as if in cognitive science, neuroscience and anthropology: A journey among robots, blacksmiths and neurons. *Theory & Psychology*, 16(6):747–759.

Aguirre, B. E. (2005). Commentary on “understanding mass panic and other collective responses to threat and disaster”: Emergency evacuations, panic, and social psychology. *Psychiatry*, 68(2):121–129.

Aguirre, B. E., El-Tawil, S., Best, E., Gill, K. B., and Fedorov, V. (2011). Contributions of social science to agent-based models of building evacuation. *Contemporary Social Science*, 6(3):415–432.

Ahmed, Q. A., Arabi, Y. M., and Memish, Z. A. (2006). Health risks at the Hajj. *The Lancet*, 367(9515):1008–1015.

Albright, T. D. and Stoner, G. R. (1995). Visual motion perception. *Proceedings of the National Academy of Sciences*, 92:2433–2440.

Allen, M. P. (1987). *Computer Simulation Of Liquids*. Oxford Science Publications. Oxford University Press, Oxford.

Alur, R., Kannan, S., and Yannakakis, M. (1999). Communicating hierarchical state machines. In Wiedermann, J., Emde Boas, P., and Nielsen, M., editors, *Automata, Languages and Programming*, volume 1644 of *Lecture Notes in Computer Science*, pages 169–178. Springer, Berlin Heidelberg.

Alur, R. and Yannakakis, M. (1998). Model checking of hierarchical state machines. In *Proceedings of the 6th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, SIGSOFT ’98/FSE-6, pages 175–188, New York, NY. ACM.

Anderson, A. E., Ellis, B. J., and Weiss, J. A. (2007). Verification, validation and sensitivity studies in computational biomechanics. *Computer Methods in Biomechanics and Biomedical Engineering*, 10(3):171–184.

Anderson, M. L. (2003). Embodied cognition: A field guide. *Artificial Intelligence*, 149(1):91–130.

Antonini, G. (2005). *A Discrete Choice Modeling Framework for Pedestrian Walking Behavior with Application to Human Tracking in Video Sequences*. PhD thesis, École polytechnique fédérale de Lausanne.

Antonini, G., Bierlaire, M., and Weber, M. (2006). Discrete choice models of pedestrian walking behavior. *Transportation Research Part B: Methodological*, 40(8):667–687.

Apache Software Foundation (2015). Apache Maven. Online: maven.apache.org. Accessed 18. December 2015.

Appert-Rolland, C., Cividini, J., and Hilhorst, H. J. (2011). Frozen shuffle update for an asymmetric exclusion process on a ring. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(7):P07009.

Arikan, O., Chenney, S., and Forsyth, D. (2001). Efficient multi-agent path planning. In Magnenat-Thalmann, N. and Thalmann, D., editors, *Computer Animation and Simulation 2001*, Eurographics, pages 151–162, Vienna. Springer.

Arkin, R. C. (1998). *Behavior-Based Robotics*. MIT Press, Cambridge, MA.

Asch, S. E. (1952). *Social psychology*. Prentice Hall, New Jersey.

Aveni, A. F. (1977). The not-so-lonely crowd: Friendship groups in collective behavior. *Sociometry*, 40(1):96–99.

Baba, Y., Tsukada, A., and Comer, C. M. (2010). Collision avoidance by running insects: antennal guidance in cockroaches. *Journal of Experimental Biology*, 213:2294–2302.

Bader, M., Bungartz, H.-J., and Weinzierl, T., editors (2013). *Advanced Computing*, volume 93 of *Lecture Notes in Computational Science and Engineering*. Springer, Berlin Heidelberg.

Bailey, R. A. (2008). *Design of Comparative Experiments*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge.

Ballerini, M., Cabibbo, N., Candelier, R., Cavagna, A., Cisbani, E., Giardina, I., Lecomte, V., Orlandi, A., Parisi, G., Procaccini, A., Viale, M., and Zdravkovic, V. (2008). Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. *Proceedings of the National Academy of Sciences*, 105(4):1232–1237.

Balzert, H. (2009). *Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering*. Springer Spektrum, Heidelberg, 3rd edition.

Balzert, H. (2011). *Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb*. Springer Spektrum, Heidelberg, 3rd edition.

Bando, M., Hasebe, K., Nakayama, A., Shibata, A., and Sugiyama, Y. (1995). Dynamical model of traffic congestion and numerical simulation. *Physical Review E*, 51:1035–1042.

Banks, C. M. and Sokolowski, J. A., editors (2009). *Principles of Modeling and Simulation: A Multidisciplinary Approach*. John Wiley & Sons, Hoboken, NJ.

Bauer, D. and Kitazawa, K. (2010). Using laser scanner data to calibrate certain aspects of microscopic pedestrian motion models. In Klingsch, W. W. F., Rogsch, C., Schadschneider, A., and Schreckenberg, M., editors, *Pedestrian and Evacuation Dynamics 2008*, pages 83–94, Berlin Heidelberg. Springer.

Bellomo, N. and Dogbe, C. (2011). On the modeling of traffic and crowds: A survey of models, speculations, and perspectives. *SIAM Review*, 53(3):409–463.

Berg, J., Guy, S. J., Lin, M., and Manocha, D. (2011). Reciprocal n-body collision avoidance. *Springer Tracts in Advanced Robotics*, 70:3–19.

Biham, O., Middleton, A. A., and Levine, D. (1992). Self-organization and a dynamical transition in traffic-flow models. *Physical Review A*, 46:R6124–R6127.

Bloch, J. (2008). *Effective Java*. Addison-Wesley, Upper Saddle River, NJ, 2nd edition.

Blue, V. J. and Adler, J. L. (2001). Cellular automata microsimulation for modeling bi-directional pedestrian walkways. *Transportation Research Part B: Methodological*, 35:293–312.

Blue, V. J., Embrechts, M. J., and Adler, J. L. (1997). Cellular automata modeling of pedestrian movements. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 2320–2323.

Boccara, N. (2010). *Modeling Complex Systems*. Graduate Texts in Physics. Springer, New York, 2nd edition.

Bocian, M., Macdonald, J., and Burn, J. (2012). Biomechanically inspired modelling of pedestrian-induced forces on laterally oscillating structures. *Journal of Sound and Vibration*, 331(16):3914–3929.

Bode, N. W. F. and Codling, E. A. (2013). Human exit route choice in virtual crowd evacuations. *Animal Behaviour*, 86(2):347–358.

Bode, N. W. F., Franks, D. W., and Wood, A. J. (2010). Limited interactions in flocks: relating model simulations to empirical data. *Journal of The Royal Society Interface*.

Bode, N. W. F., Kemloh Wagoum, A. U., and Codling, E. A. (2014). Human responses to multiple sources of directional information in virtual crowd evacuations. *Journal of The Royal Society Interface*, 11(91):20130904.

Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7280–7287.

Bornstein, M. H. and Bornstein, H. G. (1976). The pace of life. *Nature*, 259:557–559.

Boston Dynamics (2016). Atlas - the agile anthropomorphic robot. Online: www.bostondynamics.com/robot_Atlas.html. Accessed 24. February 2016.

Bouza, J. P. (2012). Planes of human anatomy. Wikimedia Commons. commons.wikimedia.org/wiki/File:Human_anatomy_planes.svg. Accessed 18. August 2014. Creative Commons Attribution 3.0 Unported (creativecommons.org/licenses/by/3.0/deed.en).

Brankov, J. G., Priezzhev, V. B., and Shelest, R. V. (2004). Generalized determinant solution of the discrete-time totally asymmetric exclusion process and zero-range process. *Physical Review E*, 69:066136.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and Regression Trees*. Chapman & Hall.

Brent, R. P. (1973). *Algorithms for Minimization without Derivatives*. Prentice-Hall, Englewood Cliffs, N.J.

Brooks, R. A. (1989). Robot that walks; emergent behaviors from a carefully evolved network. *Neural Computation*, 1(2):253–262.

Bukáček, M., Hrabák, P., and Krbálek, M. (2014). Cellular model of pedestrian dynamics with adaptive time span. In Wyrzykowski, R., Dongarra, J., Karczewski, K., and Waśniewski, J., editors, *Parallel Processing and Applied Mathematics*, volume 8385 of *Lecture Notes in Computer Science*, pages 669–678. Springer, Berlin Heidelberg.

Bungartz, H.-J., Zimmer, S., Buchholz, M., and Pflüger, D. (2014). *Modeling and Simulation: An Application-Oriented Introduction*. Springer Undergraduate Texts in Mathematics and Technology. Springer, Berlin Heidelberg.

Burstedde, C., Klauck, K., Schadschneider, A., and Zittartz, J. (2001). Simulation of pedestrian dynamics using a two-dimensional cellular automaton. *Physica A: Statistical Mechanics and its Applications*, 295:507–525.

Buschmann, T., Ewald, A., von Twickel, A., and Büschges, A. (2015). Controlling legs for locomotion—insights from robotics and neurobiology. *Bioinspiration & Biomimetics*, 10(4):041001.

Canny, J. and Reif, J. (1987). New lower bound techniques for robot motion planning problems. In *Foundations of Computer Science, 1987., 28th Annual Symposium on*, pages 49–60.

Carroll, S. P., Owen, J., and Hussein, M. F. M. (2013). A coupled biomechanical/discrete element crowd model of crowd-bridge dynamic interaction and application to the clifton suspension bridge. *Engineering Structures*, 49(0):58–75.

Cass, S. (2015). The 2015 top ten programming languages. Online on IEEE Spectrum: spectrum.ieee.org/computing/software/the-2015-top-ten-programming-languages. Accessed 16. December 2015.

Cavagna, G. A. and Margaria, R. (1966). Mechanics of walking. *Journal of Applied Physiology*, 21(1):271–278.

Chandler, R. E., Herman, R., and Montroll, E. W. (1958). Traffic dynamics: Studies in car following. *Operations Research*, 6(2):165–184.

Chattaraj, U., Seyfried, A., and Chakroborty, P. (2009). Comparison of pedestrian fundamental diagram across cultures. *Advances in Complex Systems*, 12(3):393–405.

Chraibi, M., Kemloh, U., Schadschneider, A., and Seyfried, A. (2011). Force-based models of pedestrian dynamics. *Networks and Heterogeneous Media*, 6(3):425–442.

Chraibi, M., Seyfried, A., and Schadschneider, A. (2010). Generalized centrifugal-force model for pedestrian dynamics. *Physical Review E*, 82(4):046111.

Chu, M. and Law, K. (2013). Computational framework incorporating human behaviors for egress simulations. *Journal of Computing in Civil Engineering*, 27(6):699–707.

Chu, M., Pan, X., and Law, K. (2011). Incorporating social behaviors in egress simulation. In *Computing in Civil Engineering*, chapter 67, pages 544–551. American Society of Civil Engineers.

Clark, A. (1997). *Being there: Putting brain, body and world together again*. Bradford Books. MIT Press, Cambridge, MA.

Clark, A. and Chalmers, D. (1998). The extended mind. *Analysis*, 58:7–9.

Codling, E. A., Plank, M. J., and Benhamou, S. (2008). Random walk models in biology. *Journal of The Royal Society Interface*, 5(25):813–834.

Coleman, J. S. and James, J. (1961). The equilibrium size distribution of freely-forming groups. *Sociometry*, 24(1):36–45.

Collett, T. S. and Graham, P. (2004). Animal navigation: Path integration, visual landmarks and cognitive maps. *Current Biology*, 14(12):R475–R477.

Collins, S., Ruina, A., Tedrake, R., and Wisse, M. (2005). Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307(5712):1082–1085.

Conlin, J. A. (2009). Getting around: making fast and frugal navigation decisions. In Raab, M., Johnson, J. G., and Heekeren, H. R., editors, *Mind and Motion: The Bidirectional Link between Thought and Action*, volume 174 of *Progress in Brain Research*, pages 109–117. Elsevier.

Conway, R. W., Johnson, B. M., and Maxwell, W. L. (1959). Some problems of digital systems simulation. *Management Science*, 6(1):92–110.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to algorithms*. MIT Press, Cambridge, MA, 3rd edition.

Cotterill, R. (2002). *Biophysics: An Introduction*. Wiley.

Couzin, I. D. and Krause, J. (2003). Self-organization and collective behavior in vertebrates. volume 32 of *Advances in the Study of Behavior*, pages 1–75. Academic Press.

Cundall, P. and Strack, O. (1979). A discrete numerical model for granular assemblies. *Geotechnique*, 29(1):47–65.

Curtis, S. (2013). *Pedestrian velocity obstacles: Pedestrian simulation through reasoning in velocity space*. PhD thesis, University of North Carolina at Chapel Hill.

Curtis, S., Best, A., and Manocha, D. (2015). Menge. Online: gamma.cs.unc.edu/Menge. Accessed 16. December 2015.

Curtis, S., Best, A., and Manocha, D. (2016). Menge: A modular framework for simulating crowd movement. *Collective Dynamics*. To be published.

Curtis, S. and Manocha, D. (2014). Pedestrian simulation using geometric reasoning in velocity space. In Weidmann, U., Kirsch, U., and Schreckenberg, M., editors, *Pedestrian and Evacuation Dynamics 2012*, pages 875–890. Springer International Publishing.

Custers, R. and Aarts, H. (2010). The unconscious will: How the pursuit of goals operates outside of conscious awareness. *Science*, 329(5987):47–50.

Cutting, J. E., Vishton, P. M., and Braren, P. A. (1995). How we avoid collisions with stationary and moving objects. *Psychological Review*, 102:627–651.

Davidich, M., Geiss, F., Mayer, H. G., Pfaffinger, A., and Royer, C. (2013). Waiting zones for realistic modelling of pedestrian dynamics: A case study using two major german railway stations as examples. *Transportation Research Part C: Emerging Technologies*, 37:210–222.

Davidich, M. and Köster, G. (2012). Towards automatic and robust adjustment of human behavioral parameters in a pedestrian stream model to measured data. *Safety Science*, 50(5):1253–1260.

Davidich, M. and Köster, G. (2013). Predicting pedestrian flow: A methodology and a proof of concept based on real-life data. *PLoS ONE*, 8(12):1–11.

de Oliveira, R. F., Damisch, L., Hossner, E.-J., Oudejans, R. R. D., Raab, M., Volz, K. G., and Williams, A. M. (2009). The bidirectional links between decision making, perception, and action. In Markus Raab, J. G. J. and Heekeren, H. R., editors, *Mind and Motion: The Bidirectional Link between Thought and Action*, volume 174 of *Progress in Brain Research*, pages 85–93. Elsevier.

Deuflhard, P. (1985). Recent progress in extrapolation methods for ordinary differential equations. *SIAM Review*, 27(4):505–535.

Dietrich, F., Disselnkötter, S., and Köster, G. (2015). How to get a model in pedestrian dynamics to produce stop and go waves? In *Traffic and Granular Flow '15*. Springer.

Dietrich, F. and Köster, G. (2014). Gradient navigation model for pedestrian dynamics. *Physical Review E*, 89(6):062801.

Dietrich, F., Köster, G., Seitz, M., and von Sivers, I. (2014). Bridging the gap: From cellular automata to differential equation models for pedestrian dynamics. *Journal of Computational Science*, 5(5):841–846.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271.

Dijkstra, J., Jessurun, J., and Timmermans, H. (2001). A multi-agent cellular automata model of pedestrian movement. In Schreckenberg, M. and Sharma, S. D., editors, *Pedestrian and Evacuation Dynamics*, pages 173–181. Springer.

Dijkstra, J., Jessurun, J., Vries, B., and Timmermans, H. (2006). Agent architecture for simulating pedestrians in the built environment. In *Agents in Traffic and Transportation*, pages 8–16.

Dingle, H. (2014). *Migration: The Biology of Life on the Move*. Oxford University Press, Oxford, 2nd edition.

Dingle, H. and Drake, V. A. (2007). What is migration? *BioScience*, 57(2):113–121.

Dingsøyr, T., Dybå, T., and Moe, N. B., editors (2010). *Agile Software Development: Current Research and Future Directions*. Springer, Berlin Heidelberg.

Drury, J., Cocking, C., and Reicher, S. (2009). Everyone for themselves? A comparative study of crowd solidarity among emergency survivors. *British Journal of Social Psychology*, 28:487–506.

Drury, J., Novelli, D., and Stott, C. (2013). Representing crowd behaviour in emergency planning guidance: 'mass panic' or collective resilience? *Resilience: International Policies, Practices and Discourses*, 1:18–37.

Drury, J. and Reicher, S. (1999). The intergroup dynamics of collective empowerment: Substantiating the social identity model of crowd behavior. *Group Processes & Intergroup Relations*, 2(4):381–402.

Drury, J. and Reicher, S. (2010). Collective action and psychological change: The emergence of new social identities. *British Journal of Social Psychology*, 39(4):579–604.

Drury, J. and Stott, C. (2011). Contextualising the crowd in contemporary social science. *Contemporary Social Science*, 6(3):275–288.

Duives, D. C., Daamen, W., and Hoogendoorn, S. P. (2013). State-of-the-art crowd motion simulation models. *Transportation Research Part C: Emerging Technologies*, 37(0):193–209.

Eclipse Foundation (2015). Eclipse. Online: eclipse.org. Accessed 18. December 2015.

Elliot, A. J. (2006). The hierarchical model of approach-avoidance motivation. *Motivation and Emotion*, 30(2):111–116.

Elliott, D. and Smith, D. (1993). Football stadia disasters in the united kingdom: learning from tragedy? *Organization & Environment*, 7:205–229.

Emmerich, H. and Rank, E. (1997). An improved cellular automaton model for traffic flow simulation. *Physica A: Statistical Mechanics and its Applications*, 234(3-4):676–686.

Etienne, A. S., Maurer, R., Berlie, J., Reverdin, B., Rowe, T., Georgakopoulos, J., and Seguinot, V. (1998). Navigation through vector addition. *Nature*, 396(6707):161–164.

Evans, M. R. (1997). Exact steady states of disordered hopping particle models with parallel and ordered sequential dynamics. *Journal of Physics A: Mathematical and General*, 30(16):5669.

Fajen, B. R., Parade, M. S., and Matthis, J. S. (2013). Humans perceive object motion in world coordinates during obstacle avoidance. *Journal of Vision*, 13(8):1–13.

Faria, J. J., Dyer, J. R. G., Tosh, C. R., and Krause, J. (2010). Leadership and social information use in human crowds. *Animal Behaviour*, 79(4):895–901.

FDS Evac contributors (2015). FDS+Evac website. Online: virtual.vtt.fi/virtual/proj6/fdsevac/documents/FDS+Evac_textbased_homepage.txt. Accessd 18. January 2016.

Feinberg, W. E. and Johnson, N. R. (1995). Firescap: A computer simulation model of reaction to a fire alarm. *The Journal of Mathematical Sociology*, 20(2–3):247–269.

Feliciani, C. and Nishinari, K. (2016). An improved cellular automata model to simulate the behavior of high density crowd and validation by experimental data. *Physica A: Statistical Mechanics and its Applications*, 451:135–148.

Fiorini, P. and Shiller, Z. (1993). Motion planning in dynamic environments using the relative velocity paradigm. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, volume 1, pages 560–565.

Fiorini, P. and Shiller, Z. (1998). Motion planning in dynamic environments using velocity obstacles. *The International Journal of Robotics Research*, 17(7):760–772.

Flötteröd, G. and Lämmel, G. (2015). Bidirectional pedestrian fundamental diagram. *Transportation Research Part B: Methodological*, 71(0):194 – 212.

Fraichard, T. (1993). Dynamic trajectory planning with dynamic constraints: A 'state-time space' approach. In *Intelligent Robots and Systems '93, IROS '93. Proceedings of the 1993 IEEE/RSJ International Conference on*, volume 2, pages 1393–1400.

Frigg, R. and Hartmann, S. (2012). Models in science. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Center for the Study of Language and Information, Stanford University, fall 2012 edition.

Fukagawa, N. K., Wolfson, L., Judge, J., Whipple, R., and King, M. (1995). Strength is a major factor in balance, gait, and the occurrence of falls. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 50A(Special Issue):64–67.

Fukui, M. and Ishibashi, Y. (1999). Self-organized phase transitions in cellular automaton models for pedestrians. *Journal of the Physical Society of Japan*, 68(8):2861–2863.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, Boston, MA.

Garcimartín, A., Zuriguel, I., Pastor, J. M., Martín-Gómez, C., and Parisi, D. R. (2014). Experimental evidence of the "faster is slower" effect. In *The Conference in Pedestrian and Evacuation Dynamics 2014*, Transportation Research Procedia, pages 760–767, Delft, The Netherlands.

Gazis, D. C. (2002). The origins of traffic theory. *Operations Research*, 50(1):69–77.

Gehl, J. (2010). *Cities for People*. Island Press, Washington, DC.

Gerrig, R. J. and Zimbardo, P. G. (2002). *Psychology And Life*, chapter Glossary of Psychological Terms. Allyn and Bacon, 16th edition. Published online by the American Psychological Association: http://www.apa.org/research/action/glossary.aspx. Accessed 11. April 2014.

Gigerenzer, G. (1996). On narrow norms and vague heuristics: A reply to Kahneman and Tversky. *Psychological Review*, 103(3):592–596.

Gigerenzer, G. (2008). Why heuristics work. *Perspectives on Psychological Science*, 3(1):20–29.

Gigerenzer, G. and Goldstein, D. G. (1996). Reasoning the fast and frugal way: Models of bounded rationality. *Psychological Review*, 103(4):650–669.

Gigerenzer, G. and Selten, R. (2001). Rethinking rationality. In Gigerenzer, G. and Selten, R., editors, *Bounded Rationality: The Adaptive Toolbox*, chapter 1. MIT Press, Cambridge, MA.

Gigerenzer, G. and Todd, P. M. (1999). Fast and frugal heuristics: The adpative toolbox. In *Simple Heuristics That Make Us Smart*, chapter 1, pages 3–34. Oxford University Press, Oxford.

Gigerenzer, G., Todd, P. M., and A.B.C. Research Group (1999). *Simple Heuristics That Make Us Smart*. Oxford University Press, Oxford.

Gipps, P. (1986). The role of computer graphics in validating simulation models. *Mathematics and Computers in Simulation*, 28(4):285–289.

Gipps, P. and Marksjö, B. (1985). A micro-simulation model for pedestrian flows. *Mathematics and Computers in Simulation*, 27(2–3):95–105.

Git Contributors (2015). Git. Online: `git-scm.com`. Accessed 18. December 2015.

Goldstein, D. G. and Gigerenzer, G. (2002). Models of ecological rationality: The recognition heuristic. *Psychological Review*, 109(1):75–90.

Goldstone, R. L. and Janssen, M. A. (2005). Computational models of collective behavior. *Trends in Cognitive Sciences*, 9(9):424–430.

Green, N. and Heekeren, H. R. (2009). Perceptual decision making: a bidirectional link between mind and motion. In Markus Raab, J. G. J. and Heekeren, H. R., editors, *Mind and Motion: The Bidirectional Link between Thought and Action*, volume 174 of *Progress in Brain Research*, pages 207–218. Elsevier.

Grieve, D. W. and Gear, R. J. (1966). The relationships between length of stride, step frequency, time of swing and speed of walking for children and adults. *Ergonomics*, 9(5):379–399.

Grimm, V. and Railsback, S. F. (2005). *Individual-based Modeling and Ecology*. Princeton Series in Theoretical and Computational Biology. Princeton University Press.

Guy, S. J., Chhugani, J., Curtis, S., Dubey, P., Lin, M., and Manocha, D. (2010). Pledestrian: A least-effort approach to crowd simulation. In *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*. The Eurographics Association.

Guy, S. J., Curtis, S., Lin, M. C., and Manocha, D. (2012). Least-effort trajectories lead to emergent crowd behaviors. *Physical Review E*, 85:016110.

Gwynne, S., Galea, E., Owen, M., Lawrence, P., and Filippidis, L. (1999). A review of the methodologies used in the computer simulation of evacuation from the built environment. *Building and Environment*, 34(6):741–749.

Hall, E. T. (1966). *The Hidden Dimension*. Doubleday, New York.

Hamon, A., Aoustin, Y., and Caro, S. (2014). Two walking gaits for a planar bipedal robot equipped with a four-bar mechanism for the knee joint. *Multibody System Dynamics*, 31(3):283–307.

Harel, D. (1987). Statecharts: a visual formalism for complex systems. *Science of Computer Programming*, 8(3):231–274.

Harel, D. (1988). On visual formalisms. *Communications of the ACM*, 31(5):514–530.

Hartmann, D. (2010). Adaptive pedestrian dynamics based on geodesics. *New Journal of Physics*, 12:043032.

Hartmann, D. and Hasel, P. (2014). Efficient dynamic floor field methods for microscopic pedestrian crowd simulations. *Communications in Computational Physics*, 16(1):264–286.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer Series in Statistics. Springer, New York.

Hastorf, A. H. and Cantril, H. (1954). They saw a game; a case study. *The Journal of Abnormal and Social Psychology*, 49(1):129–134.

Hatze, H. (1974). The meaning of the term 'biomechanics'. *Journal of Biomechanics*, 7(2):189–190.

Helbing, D. (1991). A mathematical model for the behavior of pedestrians. *Behavioral Science*, 36(4):298–310.

Helbing, D. (1993). Stochastic and boltzmann-like models for behavioral changes, and their relation to game theory. *Physica A: Statistical Mechanics and its Applications*, 193(2):241–258.

Helbing, D., Buzna, L., Johansson, A., and Werner, T. (2005). Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions. *Transportation Science*, 39(1):1–24.

Helbing, D., Farkas, I., and Vicsek, T. (2000a). Simulating dynamical features of escape panic. *Nature*, 407:487–490.

Helbing, D., Farkas, I. J., and Vicsek, T. (2000b). Freezing by heating in a driven mesoscopic system. *Physical Review Letters*, 84:1240–1243.

Helbing, D., Johansson, A., and Al-Abideen, H. Z. (2007). Dynamics of crowd disasters: An empirical study. *Physical Review E*, 4(75):046109.

Helbing, D. and Molnár, P. (1995). Social Force Model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286.

Helbing, D., Molnár, P., Farkas, I. J., and Bolay, K. (2001). Self-organizing pedestrian movement. *Environment and Planning B: Planning and Design*, 28:361–383.

Hinkelmann, K. and Kempthorne, O. (2008). *Design and Analysis of Experiments: Volume 1, Introduction to Experimental Design.* Wiley, Hoboken, NJ, 2nd edition.

Höcker, M., Berkhahn, V., Kneidl, A., Borrmann, A., and Klein, W. (2010). Graph-based approaches for simulating pedestrian dynamics in building models. In Scherer, R., editor, *Proceedings of the European Conference on Product and Process Modelling 2010*, pages 389–394, Cork, Republic of Ireland. CRC Press.

Honda Motor Co., Ltd. (2004). Honda ASIMO robot fact sheet. Online: asimo.honda.com/downloads/pdf/honda-asimo-robot-fact-sheet.pdf. Accessed 18. August 2014.

Hoogendoorn, S. P. (2007). Pedestrian flow modeling by adaptive control. *Transportation Research Record*, 1878:95–103.

Hoogendoorn, S. P. and Bovy, P. H. L. (2003). Simulation of pedestrian flows by optimal control and differential games. *Optimal Control Applications and Methods*, 24(3):153–172.

Hoogendoorn, S. P. and Bovy, P. H. L. (2004). Pedestrian route-choice and activity scheduling theory and models. *Transportation Research Part B: Methodological*, 38(2):169–190.

Hunt, K. D. (1994). The evolution of human bipedality: ecology and functional morphology. *Journal of Human Evolution*, 26(3):183–202.

Hutchinson, J. M. C. and Gigerenzer, G. (2005). Simple heuristics and rules of thumb: Where psychologists and behavioural biologists might meet. *Behavioural Processes*, 69(2):97–124. Proceedings of the meeting of the Society for the Quantitative Analyses of Behavior (SQAB 2004).

Iida, F., Rummel, J., and Seyfarth, A. (2007). Bipedal walking and running with compliant legs. In *2007 IEEE International Conference on Robotics and Automation*, pages 3970–3975.

Ioannidis, J. P. A. (2005). Why most published research findings are false. *PLOS Medicine*, 2(8):e124.

Iosa, M., Fusco, A., Morone, G., and Paolucci, S. (2012). Walking there: Environmental influence on walking-distance estimation. *Behavioural Brain Research*, 226(1):124–132.

James, J. (1953). The distribution of free-forming small group size. *American Sociological Review*, 18(5):569–570.

Jelić, A., Appert-Rolland, C., Lemercier, S., and Pettré, J. (2012a). Properties of pedestrians walking in line: Fundamental diagrams. *Physical Review E*, 85(3):036111.

Jelić, A., Appert-Rolland, C., Lemercier, S., and Pettré, J. (2012b). Properties of pedestrians walking in line. ii. stepping behavior. *Physical Review E*, 86(4):046111.

Johansson, A., Helbing, D., and Shukla, P. (2007). Specification of the social force pedestrian model by evolutionary adjustment to video tracking data. *Advances in Complex Systems*, 10:271–288.

Johansson, F., Peterson, A., and Tapani, A. (2015). Waiting pedestrians in the social force model. *Physica A: Statistical Mechanics and its Applications*, 419:95–107.

Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2):201–211.

Johnson, N. R. (1987). Panic and the breakdown of social order: Popular myth, social theory, empirical evidence. *Sociological Focus*, 20(3):171–183.

Jovancevic, J., Sullivan, B., and Hayhoe, M. (2006). Control of attention and gaze in complex environments. *Journal of Vision*, 6(12):1431–1450.

JSON Contributors (2015). JSON (JavaScript Object Notation). Online: `www.json.org`. Accessed 18. December 2015.

JuPedSim Contributors (2015). Jupedsim. Online: `www.fz-juelich.de/ias/jsc/EN/Research/ModellingSimulation/CivilSecurityTraffic/PedestrianDynamics/Activities/JuPedSim/jupedsimNode.html`. Accessed 16. December 2015.

Kahneman, D. (2012). *Thinking, Fast and Slow*. Penguin Books, London.

Kahneman, D. and Tversky, A. (1996). On the reality of cognitive illusions. *Psychological Review*, 103:582–591.

Kant, K. and Zucker, S. W. (1986). Toward efficient trajectory planning: The path-velocity decomposition. *The International Journal of Robotics Research*, 5(3):72–89.

Kelley, H. H., Jr., J. C. C., Dahlke, A. E., and Hill, A. H. (1965). Collective behavior in a simulated panic situation. *Journal of Experimental Social Psychology*, 1(1):20–54.

Kelman, M. (2011). *The Heuristics Debate*. Oxford University Press, Oxford.

Kielar, P. M. and Borrmann, A. (2016). Modeling pedestrians' interest in locations: A concept to improve simulations of pedestrian destination choice. *Simulation Modelling Practice and Theory*, 61:47–62.

Kielar, P. M., Handel, O., Biedermann, D. H., and Borrmann, A. (2014). Concurrent hierarchical finite state machines for modeling pedestrian behavioral tendencies. In *The Conference in Pedestrian and Evacuation Dynamics 2014*, Transportation Research Procedia, pages 576–584, Delft, The Netherlands.

Kirik, E. S., Yurgelyan, T. B., and Krouglov, D. V. (2009). The shortest time and/or the shortest path strategies in a CA FF pedestrian dynamics model. *Mathematics and Physics*, 2(3):271–278.

Kirtley, C., Whittle, M. W., and Jefferson, R. (1985). Influence of walking speed on gait parameters. *Journal of Biomedical Engineering*, 7(4):282–288.

Klein, W., Köster, G., and Meister, A. (2010). Towards the calibration of pedestrian stream models. In Wyrzykowski, R., Dongarra, J., Karczewski, K., and Wasniewski, J., editors, *Parallel Processing and Applied Mathematics*, volume 6068 of *Lecture Notes in Computer Science*, pages 521–528. Springer.

Kleinert, J., Obermayr, M., and Balzer, M. (2013). Modeling of large scale granular systems using the discrete element method and the non-smooth contact dynamics method: A comparison. Technical Report 238 (2013), Fraunhofer-Institut für Techno- und Wirtschaftsmathematik (Fraunhofer ITWM).

Klüpfel, H. L. (2003). *A Cellular Automaton Model for Crowd Movement and Egress Simulation*. PhD thesis, Universität Duisburg-Essen.

Kneidl, A. (2013). *Methoden zur Abbildung menschlichen Navigationsverhaltens bei der Modellierung von Fußgängerströmen*. PhD thesis, Technische Universität München.

Kneidl, A., Borrmann, A., and Hartmann, D. (2012). Generation and use of sparse navigation graphs for microscopic pedestrian simulation models. *Advanced Engineering Informatics*, 26(4):669–680.

Kneidl, A. and Sesser, F. (2015). accu:rate Website. Online: www.accu-rate.de. Accessed 23. December 2015.

Knudson, D. (2007). *Fundamentals of Biomechanics*. Springer, New York, 2nd edition.

Kohavi, Z. and Jha, N. K. (2010). *Switching and Finite Automata Theory*. Cambridge University Press, Cambridge, 3rd edition.

Koopman, B., Grootenboer, H. J., and de Jongh, H. J. (1995). An inverse dynamics model for the analysis, reconstruction and prediction of bipedal walking. *Journal of Biomechanics*, 28(11):1369–1376.

Köster, G., Hartmann, D., and Klein, W. (2011a). Microscopic pedestrian simulations: From passenger exchange times to regional evacuation. In Hu, B., Morasch, K., Pickl, S., and Siegle, M., editors, *Operations Research Proceedings 2010: Selected Papers of the Annual International Conference of the German Operations Research Society*, pages 571–576. Springer.

Köster, G., Lehmberg, D., and Dietrich, F. (2015). Is slowing down enough to model movement on stairs? In *Traffic and Granular Flow '15*, Nootdorp, the Netherlands. 27–30 October 2015.

Köster, G., Seitz, M., Treml, F., Hartmann, D., and Klein, W. (2011b). On modelling the influence of group formations in a crowd. *Contemporary Social Science*, 6(3):397–414.

Köster, G., Treml, F., and Gödel, M. (2013). Avoiding numerical pitfalls in social force models. *Physical Review E*, 87(6):063305.

Köster, G., Treml, F., Seitz, M., and Klein, W. (2014). Validation of crowd models including social groups. In Weidmann, U., Kirsch, U., and Schreckenberg, M., editors, *Pedestrian and Evacuation Dynamics 2012*, pages 1051–1063. Springer International Publishing.

Köster, G. and Zönnchen, B. (2014). Queuing at bottlenecks using a dynamic floor field for navigation. In *The Conference in Pedestrian and Evacuation Dynamics 2014*, Transportation Research Procedia, pages 344–352, Delft, The Netherlands.

Köster, G. and Zönnchen, B. (2015). A queuing model based on social attitudes. In *Traffic and Granular Flow '15*. Springer.

Kretz, T. (2007). *Pedestrian Traffic - Simulation and Experiments*. PhD thesis, Universität Duisburg Essen.

Kretz, T. (2009). Pedestrian traffic: on the quickest path. *Journal of Statistical Mechanics: Theory and Experiment*, 2009(03):P03012.

Kretz, T., Große, A., Hengst, S., Kautzsch, L., Pohlmann, A., and Vortisch, P. (2011). Quickest paths in simulations of pedestrians. *Advances in Complex Systems*, 10:733–759.

Kretz, T., Grünebohm, A., Kaufman, M., Mazur, F., and Schreckenberg, M. (2006a). Experimental study of pedestrian counterflow in a corridor. *Journal of Statistical Mechanics: Theory and Experiment*, 2006(10):P10001.

Kretz, T., Wölki, M., and Schreckenberg, M. (2006b). Characterizing correlations of flow oscillations at bottlenecks. *Journal of Statistical Mechanics: Theory and Experiment*, 2006(02):P02005.

Kuo, A. D. (2001). A simple model of bipedal walking predicts the preferred speed-step length relationship. *Journal of Biomechanical Engineering*, 123(3):264–269.

Kuo, A. D. (2007). The six determinants of gait and the inverted pendulum analogy: A dynamic walking perspective. *Human Movement Science*, 26(4):617–656.

Lacquaniti, F., Grasso, R., and Zago, M. (1999). Motor patterns in walking. *Physiology*, 14(4):168–174.

Lakoba, T. I., Kaup, D. J., and Finkelstein, N. M. (2005). Modifications of the helbing-molnár-farkas-vicsek social force model for pedestrian evolution. *Simulation*, 81(5):339–352.

Langrock, R., Hopcraftzh, J. G. C., Blackwell, P. G., Goodall, V., King, R., Niu, M., Patterson, T. A., Pedersen, M. W., Skarin, A., and Schick, R. S. (2014). Modelling group dynamic animal movement. *Methods in Ecology and Evolution*, 5(2):190–199.

Langston, P. A., Masling, R., and Asmar, B. N. (2006). Crowd dynamics discrete element multi-circle model. *Safety Science*, 44(5):395–417.

Le Bon, G. (1996). *The Crowd: A Study of the Popular Mind*. Project Gutenberg. Available online: www.gutenberg.org/ebooks/445.

Lehto, A., Morello, D., and Korppoo, K. (2015). Game design deep dive: Traffic systems in *Cities: Skylines*. Online: Gamasutra, www.gamasutra.com/view/news/239534. Accessed 11. March 2016.

Leng, B., Wang, J., Zhao, W., and Xiong, Z. (2014). An extended floor field model based on regular hexagonal cells for pedestrian simulation. *Physica A: Statistical Mechanics and its Applications*, 402:119–133.

Levine, R. V. and Norenzayan, A. (1999). The pace of life in 31 countries. *Journal of Cross-Cultural Psychology*, 30(2):178–205.

Lewin, K. (1951). *Field theory in social science: Selected theoretical papers*. Harper, New York.

Leyden, K. M. (2003). Social capital and the built environment: The importance of walkable neighborhoods. *American Journal of Public Health*, 93(9):1546–1551.

Liddle, J., Seyfried, A., Klingsch, W., Rupprecht, T., Schadschneider, A., and Winkens, A. (2009). An experimental study of pedestrian congestions: Influence of bottleneck width and length. *arXiv*, 0911.4350(v2).

Liddle, J., Seyfried, A., Steffen, B., Klingsch, W., Rupprecht, T., Winkens, A., and Boltes, M. (2011). Microscopic insights into pedestrian motion through a bottleneck, resolving spatial and temporal variations. *arXiv*, 1105.1532(v1).

Macdonald, J. H. (2009). Lateral excitation of bridges by balancing pedestrians. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 465:1055–1073.

Maniccam, S. (2003). Traffic jamming on hexagonal lattice. *Physica A: Statistical Mechanics and its Applications*, 321(3–4):653–664.

Manning, A. and Dawkins, M. S. (2012). *An Introduction to Animal Behaviour*. Cambridge University Press, Cambridge, 6th edition.

Martin, A. E. and Schmiedeler, J. P. (2014). Predicting human walking gaits with a simple planar model. *Journal of Biomechanics*, 47(6):1416–1421.

MATSim Contributors (2015). Matsim. Online: matsim.org. Accessed 16. December 2015.

Matsumoto, D. (2012). The psychological dimensions of context. *Acta de Investigación Psicológica*, 2(2):611–622.

McGeer, T. (1993). Dynamics and control of bipedal locomotion. *Journal of Theoretical Biology*, 163(3):277–314.

McLeod, P. and Dienes, Z. (1996). Do fielders know where to go to catch the ball or only how to get there? *Journal of Experimental Psychology: Human Perception and Performance*, 22(3):531–543.

McNaughton, B. L., Battaglia, F. P., Jensen, O., Moser, E. I., and Moser, M.-B. (2006). Path integration and the neural basis of the 'cognitive map'. *Nature Reviews Neuroscience*, 7(8):663–678.

McNeill Alexander, R. (2003). *Principles of Animal Locomotion*. Princeton University Press, Princeton, N.J.

Millington, I. and Funge, J. (2009). *Artificial Intelligence for Games*. Morgan Kaufmann.

Mitarai, N. and Nakanishi, H. (1999). Stability analysis of optimal velocity model for traffic and granular flow under open boundary condition. *Journal of the Physical Society of Japan*, 68(8):2475–2478.

Mittelstaedt, M.-L. and Mittelstaedt, H. (1980). Homing by path integration in a mammal. *Naturwissenschaften*, 67(11):566–567.

Molnár, P. (1996). *Modellierung und Simulation der Dynamik von Fußgängerströmen*. PhD thesis, Universität Stuttgart.

Moussaïd, M., Garnier, S., Theraulaz, G., and Helbing, D. (2009a). Collective information processing and pattern formation in swarms, flocks, and crowds. *Topics in Cognitive Science*, 1(3):469–497.

Moussaïd, M., Guillot, E. G., Moreau, M., Fehrenbach, J., Chabiron, O., Lemercier, S., Pettré, J., Appert-Rolland, C., Degond, P., and Theraulaz, G. (2012). Traffic instabilities in self-organized pedestrian crowds. *PLoS Computational Biology*, 8(3):e1002442.

Moussaïd, M., Helbing, D., Garnier, S., Johansson, A., Combe, M., and Theraulaz, G. (2009b). Experimental study of the behavioural mechanisms underlying self-organization in human crowds. *Proceedings of the Royal Society B: Biological Sciences*, 276:2755–2762.

Moussaïd, M., Helbing, D., and Theraulaz, G. (2011). How simple rules determine pedestrian behavior and crowd disasters. *Proceedings of the National Academy of Sciences*, 108(17):6884–6888.

Moussaïd, M. and Nelson, J. D. (2014). Simple heuristics and the modelling of crowd behaviours. In Weidmann, U., Kirsch, U., and Schreckenberg, M., editors, *Pedestrian and Evacuation Dynamics 2012*, pages 75–90. Springer International Publishing.

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

Muramatsu, M. and Nagatani, T. (2000). Jamming transition in two-dimensional pedestrian traffic. *Physica A: Statistical Mechanics and its Applications*, 275(1–2):281–291.

Nagel, K. and Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. *Journal de Physique I*, 2(12):2221–2229.

Nagel, K., Wolf, D. E., Wagner, P., and Simon, P. (1998). Two-lane traffic rules for cellular automata: A systematic approach. *Physical Review E*, 58:1425–1437.

Nakanishi, K. (2000). Multibunch solutions of the differential-difference equation for traffic flow. *Physical Review E*, 62:3349–3355.

Navin, F. P. and Wheeler, R. J. (1969). Pedestrian flow characteristics. *Traffic Engineering*, 19(7):30–33.

Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7:308–313.

Newell, G. F. (1961). Nonlinear effects in the dynamics of car following. *Operations Research*, 9(2):209–229.

Ngai, K. M., Burkle, F. M., Hsu, A., and Hsu, E. B. (2009). Human stampedes: A systematic review of historical and peer-reviewed sources. *Disaster Medicine and Public Health Preparedness*, 3(4):191–195.

Nisbett, R. E. and Wilson, T. D. (1977). Telling more than we can know: Verbal reports on mental processes. *Psychological Review*, 84:231–259.

Nishinari, K., Fukui, M., and Schadschneider, A. (2004). A stochastic cellular automaton model for traffic flow with multiple metastable states. *Journal of Physics A: Mathematical and General*, 37(9):3101.

Novelli, D., Drury, J., and Reicher, S. (2010). Come together: Two studies concerning the impact of group relations on personal space. *British Journal of Social Psychology*, 49(2):223–236.

Oberheim, E. and Hoyningen-Huene, P. (2013). The incommensurability of scientific theories. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Center for the Study of Language and Information, Stanford University, spring 2013 edition. Published online: plato.stanford.edu/archives/spr2013/entries/incommensurability/.

Oberkampf, W. L. and Roy, C. J. (2010). *Verification and Validation in Scientific Computing*. Cambridge University Press, Cambridge.

Ondřej, J., Pettré, J., Olivier, A.-H., and Donikian, S. (2010). A synthetic-vision based steering approach for crowd simulation. *ACM Transactions on Graphics*, 29(4):123:1–123:9.

Oracle (2015a). Java SE documentation. Online: www.oracle.com/technetwork/java/javase/documentation/index.html. Accessed 16. December 2015.

Oracle (2015b). Java VisualVM. Online: visualvm.java.net. Accessed 18. December 2015.

Pan, X., Han, C., Dauber, K., and Law, K. (2007). A multi-agent based framework for the simulation of human and social behaviors during emergency evacuations. *AI & Society*, 22:113–132.

Papadimitriou, E., Yannis, G., and Golias, J. (2009). A critical assessment of pedestrian behaviour models. *Transportation Research Part F: Traffic Psychology and Behaviour*, 12(3):242–255.

Paradox Interactive (2015). *Cities: Skylines*. Video Game.

Parisi, D. R. and Dorso, C. O. (2007). Morphological and dynamical aspects of the room evacuation process. *Physica A: Statistical Mechanics and its Applications*, 385(1):343–355.

Parisi, D. R., Gilman, M., and Moldovan, H. (2009). A modification of the social force model can reproduce experimental data of pedestrian flows in normal conditions. *Physica A: Statistical Mechanics and its Applications*, 388(17):3600–3608.

Pastor, J. M., Garcimartín, A., Gago, P. A., Peralta, J. P., Martín-Gómez, C., Ferrer, L. M., Maza, D., Parisi, D. R., Pugnaloni, L. A., and Zuriguel, I. (2015). Experimental proof of faster-is-slower in systems of frictional particles flowing through constrictions. *Physical Review E*, 92:062817.

Patton, J. W. (2007). A pedestrian world: Competing rationalities and the calculation of transportation change. *Environment and Planning A*, 39(4):928–944.

PEDSIM Contributors (2015). Pedsim. Online: pedsim.silmaril.org. Accessed 16. December 2015.

Pelechano, N., Allbeck, J. M., and Badler, N. I. (2008). *Virtual Crowds: Methods, Simulation, and Control*, volume 3 of *Synthesis Lectures on Computer Graphics and Animation*. Morgan & Claypool Publishers.

Pellegrini, S., Ess, A., Schindler, K., and Van Gool, L. (2009). You'll never walk alone: Modeling social behavior for multi-target tracking. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 261–268. IEEE.

Pellegrini, S., Ess, A., Tanaskovic, M., and Van Gool, L. (2010). Wrong turn - no dead end: A stochastic pedestrian motion model. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 15–22. IEEE.

Popper, K. (2002). *The Logic of Scientific Discovery (1934, 1959)*. Routledge Classics, London and New York.

Pöschel, T. (1994). Recurrent clogging and density waves in granular material flowing through a narrow pipe. *Journal de Physique I France*, 4(4):499–506.

Pottinger, D. (1999a). Coordinated unit movement. Online: Gamasutra, www.gamasutra.com/view/feature/131720/coordinated_unit_movement.php. Accessed 02. February 2016.

Pottinger, D. (1999b). Implementing coordinated movement. Online: Gamasutra, www.gamasutra.com/view/feature/131721. Accessed 02. February 2016.

Pratt, J., Chew, C.-M., Torres, A., Dilworth, P., and Pratt, G. (2001). Virtual model control: An intuitive approach for bipedal locomotion. *The International Journal of Robotics Research*, 20(2):129–143.

Pucher, J. and Dijkstra, L. (2000). Making walking and cycling safer: lessons from europe. *Transportation Quarterly*, 54(3):25–50.

Qiu, F. and Hu, X. (2010). Modeling group structures in pedestrian crowd simulation. *Simulation Modelling Practice and Theory*, 18(2):190–205.

Raab, M., Johnson, J. G., and Heekeren, H. R., editors (2009). *Mind and Motion: The Bidirectional Link between Thought and Action*. Progress in Brain Research. Elsevier.

Rao, K. K. and Nott, P. R. (2008). *An Introduction to Granular Flow*. Cambridge Series in Chemical Engineering. Cambridge University Press, Cambridge.

Rapoport, A. (1977). *Human Aspects of Urban Form: Towards a Man Environment Approach to Urban Form and Design*. Pergamon Press, Oxford.

Reeves, W. T. (1983). Particle systems—a technique for modeling a class of fuzzy objects. *ACM Transactions on Graphics (TOG)*, 2(2):91–108.

Reicher, S. D. (1996). 'The Battle of Westminster': developing the social identity model of crowd behaviour in order to explain the initiation and development of collective conflict. *European Journal of Social Psychology*, 26(1):115–134.

Reuter, V., Bergner, B. S., Köster, G., Seitz, M., Treml, F., and Hartmann, D. (2014). On modeling groups in crowds: Empirical evidence and simulation results including large groups. In Weidmann, U., Kirsch, U., and Schreckenberg, M., editors, *Pedestrian and Evacuation Dynamics 2012*, pages 835–845. Springer International Publishing.

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 21(4):25–34.

Reynolds, C. W. (1999). Steering behaviors for autonomous characters. In *Game Developers Conference*, pages 763–782, San Jose, CA. Miller Freeman Game Group, San Francisco, CA.

Richmond, B. G., Begun, D. R., and Strait, D. S. (2001). Origin of human bipedalism: The knuckle-walking hypothesis revisited. *American Journal of Physical Anthropology*, 116:70–105.

Rickert, M., Nagel, K., Schreckenberg, M., and Latour, A. (1996). Two lane traffic simulations using cellular automata. *Physica A: Statistical Mechanics and its Applications*, 231(4):534–550.

Rieskamp, J. and Hoffrage, U. (1999). When do people use simple heuristics, and how can we tell? In *Simple Heuristics That Make Us Smart*, chapter 7, pages 141–167. Oxford University Press, Oxford.

RiMEA (2009). *Richtlinie für Mikroskopische Entfluchtungsanalysen - RiMEA*. RiMEA e.V., 2.2.1 edition. www.rimea.de.

Robinson, A. P. and Froese, R. E. (2004). Model validation using equivalence tests. *Ecological Modelling*, 176(3–4):349–358.

Robinson, S. (2004). *Simulation: The Practice of Model Development and Use*. John Wiley & Sons.

Ronchi, E., Kuligowski, E. D., Reneke, P. A., Peacock, R. D., and Nilsson, D. (2013). The process of verification and validation of building fire evacuation models. Technical Note 1822, National Institute of Standards and Technology (NIST), U. S. Department of Commerce.

Ruesch, J. and Kees, W. (1956). *Nonverbal Communication: Notes on the Visual Perception of Human Relations*. University of California Press, Berkley and Los Angeles.

Rushdi, K., Koop, D., and Wu, C. Q. (2014). Experimental studies on passive dynamic bipedal walking. *Robotics and Autonomous Systems*, 62(4):446–455.

Russell, S. J. and Norvig, P. (2010). *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River, NJ, 3rd edition.

Schack, T. and Ritter, H. (2009). The cognitive nature of action – functional links between cognitive psychology, movement science, and robotics. In Markus Raab, J. G. J. and Heekeren, H. R., editors, *Mind and Motion: The Bidirectional Link between Thought and Action*, volume 174 of *Progress in Brain Research*, pages 231–250. Elsevier.

Schadschneider, A., Klingsch, W., Klüpfel, H., Kretz, T., Rogsch, C., and Seyfried, A. (2009). Evacuation dynamics: Empirical results, modeling and applications. In Meyers, R. A., editor, *Encyclopedia of Complexity and Systems Science*, pages 3142–3176. Springer, New York.

Schadschneider, A. and Schreckenberg, M. (1993). Cellular automation models and traffic flow. *Journal of Physics A: Mathematical and General*, 26(15):L679.

Schadschneider, A. and Seyfried, A. (2011). Empirical results for pedestrian dynamics and their implications for modeling. *Networks and Heterogeneous Media*, 6(3):545–560.

Schiff, W. and Detwiler, M. L. (1979). Information used in judging impending collision. *Perception*, 8(6):647–658.

Schiff, W. and Oldak, R. (1990). Accuracy of judging time to arrival: Effects of modality, trajectory, and gender. *Journal of Experimental Psychology: Human Perception and Performance*, 16(2):303–316.

Schmitt, D. (2003). Insights into the evolution of human bipedalism from experimental studies of humans and other primates. *Journal of Experimental Biology*, 206(9):1437–1448.

Schneider, B. (2011). *Die Simulation menschlichen Panikverhaltens: Ein Agentenbasierter Ansatz*. Vieweg+Teubner, Wiesbaden.

Schreckenberg, M., Schadschneider, A., Nagel, K., and Ito, N. (1995). Discrete stochastic models for traffic flow. *Physical Review E*, 51:2939–2949.

Searle, J. R. (2002). *Consciousness and Language*, chapter Collective Intentions and Actions, pages 90–105. Cambridge University Press, Cambridge.

Seer, S. (2015). *A unified framework for evaluating microscopic pedestrian simulation models*. PhD thesis, Technische Universität Wien, Fakultät für Mathematik und Geoinformation, Institut für Analysis und Scientific Computing. TU Wien Online Katalog (AC12656133).

Seer, S., Brändle, N., and Ratti, C. (2014). Kinects and human kinetics: A new approach for studying pedestrian behavior. *Transportation Research Part C: Emerging Technologies*, 48(0):212–228.

Seitz, M., Köster, G., and Hartmann, D. (2011). On modeling the separation and reunion of social groups. In *Proceedings of the International Conference on Emergency Evacuation of People from Buildings*, Warzaw, Poland.

Seitz, M., Köster, G., and Pfaffinger, A. (2014a). Pedestrian group behavior in a cellular automaton. In Weidmann, U., Kirsch, U., and Schreckenberg, M., editors, *Pedestrian and Evacuation Dynamics 2012*, pages 807–814. Springer International Publishing.

Seitz, M. J., Bode, N., and Köster, G. (submitted 2015a). How cognitive heuristics can explain social interactions in spatial movement. *Royal Society Interface*. Submitted.

Seitz, M. J., Dietrich, F., and Köster, G. (2014b). A study of pedestrian stepping behaviour for crowd simulation. In *The Conference in Pedestrian and Evacuation Dynamics 2014*, Transportation Research Procedia, pages 282–290, Delft, The Netherlands.

Seitz, M. J., Dietrich, F., and Köster, G. (2015b). The effect of stepping on pedestrian trajectories. *Physica A: Statistical Mechanics and its Applications*, 421:594–604.

Seitz, M. J., Dietrich, F., Köster, G., and Bungartz, H.-J. (2016). The superposition principle: A conceptual perspective on pedestrian stream simulations. *Collective Dynamics*. To be published.

Seitz, M. J. and Köster, G. (2012). Natural discretization of pedestrian movement in continuous space. *Physical Review E*, 86(4):046108.

Seitz, M. J. and Köster, G. (2014). How update schemes influence crowd simulations. *Journal of Statistical Mechanics: Theory and Experiment*, 7:P07002.

Seitz, M. J., Seer, S., Klettner, S., Köster, G., and Handel, O. (2015c). How do we wait? Fundamentals, characteristics, and modeling implications. In *Traffic and Granular Flow '15*, Nootdorp, the Netherlands. 27–30 October 2015.

Seitz, M. J., Templeton, A., Drury, J., Köster, G., and Philippides, A. (submitted 2015d). Parsimony versus reductionism: How can crowd psychology be introduced to computer simulation? *Theory & Psychology*. Submitted.

Sethian, J. A. (1996). A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595.

Sethian, J. A. (1999). *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press, Cambridge.

Seyfried, A., Steffen, B., Klingsch, W., and Boltes, M. (2005). The fundamental diagram of pedestrian movement revisited. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(10):P10002.

Shiller, Z., Large, F., and Sekhavat, S. (2001). Motion planning in dynamic environments: obstacles moving along arbitrary trajectories. In *IEEE International Conference on Robotics and Automation, 2001*, volume 4, pages 3716–3721.

Shoham, Y. and Leyton-Brown, K. (2008). *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, Cambridge.

Sime, J. D. (1995). Crowd psychology and engineering. *Safety Science*, 21(1):1–14.

Simon, H. A. (1955). A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69(1):99–118.

Simon, H. A. (1959). Theories of decision-making in economics and behavioral science. *The American Economic Review*, 49(3):253–283.

Simon, H. A. (1990). Invariants of human behavior. *Annual Review of Psychology*, 41(1):1–20.

Simon, H. A. (1996). *The Sciences of the Artificial*. MIT Press, Cambridge, MA.

Singh, H., Arter, R., Dodd, L., Langston, P., Lester, E., and Drury, J. (2009). Modelling subgroup behaviour in crowd dynamics DEM simulation. *Applied Mathematical Modelling*, 33(12):4408–4423.

Smith, A., James, C., Jones, R., Langston, P., Lester, E., and Drury, J. (2009). Modelling contra-flow in crowd dynamics DEM simulation. *Safety Science*, 47(3):395–404.

Smith, R. C. (2014). *Uncertainty Quantification: Theory, Implementation, and Applications*. Computational Science and Engineering. Society for Industrial and Applied Mathematics.

Southworth, M. (2005). Designing the walkable city. *Journal of Urban Planning and Development*, 131(4):246–257.

Starbuck, W. H. (1963). Level of aspiration theory and economic behavior. *Behavioral Science*, 8(2):128–136.

Steffen, B. and Seyfried, A. (2010). Methods for measuring pedestrian density, flow, speed and direction with minimal scatter. *Physica A: Statistical Mechanics and its Applications*, 389(9):1902–1910.

Stenning, K. and van Lambalgen, M. (2008). *Human Reasoning and Cognitive Science*. MIT Press, Cambridge, MA.

Steptoe, W., Wolff, R., Murgia, A., Guimaraes, E., Rae, J., Sharkey, P., Roberts, D., and Steed, A. (2008). Eye-tracking for avatar eye-gaze and interactional analysis in immersive collaborative virtual environments. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, pages 197–200, New York, NY. ACM.

Stott, C. and Reicher, S. (1998). Crowd action as intergroup process: introducing the police perspective. *European Journal of Social Psychology*, 28(4):509–529.

Strandburg-Peshkin, A., Twomey, C. R., Bode, N. W., Kao, A. B., Katz, Y., Ioannou, C. C., Rosenthal, S. B., Torney, C. J., Wu, H. S., Levin, S. A., and Couzin, I. D. (2013). Visual sensory networks and effective information transfer in animal groups. *Current Biology*, 23(17):R709–R711.

Strang, G. (2007). *Computational Science and Engineering*. Wellesley-Cambridge Press.

Sugiyama, Y. (1999). Optimal velocity model for traffic flow. *Computer Physics Communications*, 121–122:399–401.

SUMO Contributors (2015). SUMO – Simulation of Urban MObility. Online: www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/. Accessed 11. January 2016.

Sumpter, D. J. T. (2006). The principles of collective animal behaviour. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 361(1465):5–22.

Tajima, Y. and Nagatani, T. (2001). Scaling behavior of crowd flow outside a hall. *Physica A: Statistical Mechanics and its Applications*, 292(1–4):545–554.

Templeton, A., Drury, J., and Philippides, A. (2015). From mindless masses to small groups: Conceptualizing collective behavior in crowd modeling. *Review of General Psychology*, 19(3):215–229.

Thalmann, D. and Musse, S. R. (2012). *Crowd Simulation*. Springer, London, 2nd edition.

Thorpe, S. K. S., Holder, R. L., and Crompton, R. H. (2007). Origin of human bipedalism as an adaptation for locomotion on flexible branches. *Science*, 316(5829):1328–1331.

Tiwari, G. (2003). Transport and land-use policies in delhi. *Bulletin of the World Health Organization*, 81(6):444–450.

To, K., Lai, P.-Y., and Pak, H. K. (2001). Jamming of granular flow in a two-dimensional hopper. *Physical Review Letters*, 86:71–74.

Todd, P. M. and Gigerenzer, G. (2000). Précis of simple heuristics that make us smart. *Behavioral and Brain Sciences*, 23(5):727–741.

Torchiani, C., Seitz, M. J., Willems, D., Ruzika, S., and Köster, G. (2015). Fahrgastwechselzeiten von Shuttlebussen: Feldbeobachtung, statistische Auswertung und weitere Verwendung der Daten (Beobachtung vom 06.12.2014 in Kaiserslautern). Technical Report TUM-I1517, Technische Universität München.

Tordeux, A., Chraibi, M., and Seyfried, A. (2015). Collision-free first order model for pedestrian dynamics. In *Traffic and Granular Flow '15*, Nootdorp, the Netherlands. 27–30 October 2015.

Tordeux, A. and Seyfried, A. (2014). Collision-free nonuniform dynamics within continuous optimal velocity models. *Physical Review E*, 90:042812.

Turner, J. C. (1982). Towards a cognitive redefinition of the social group. In Tajfel, H., editor, *Social identity and intergroup relations*, pages 15–40. Cambridge University Press, Cambridge.

Turner, J. C., Oakes, P. J., Haslam, S. A., and McGarty, C. (1994). Self and collective: Cognition and social context. *Personality and Social Psychology Bulletin*, 20:454–463.

Tversky, A. and Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science*, 185:1124–1131.

Tversky, A. and Kahneman, D. (1981). The framing of decisions and the psychology of choice. *Science*, 211:435–458.

Ulicny, B. and Thalmann, D. (2002). Towards interactive real-time crowd behavior simulation. *Computer Graphics Forum*, 21(4):767–775.

University of Wuppertal (2015). Data set: Bottleneck (flow, density, velocity). Online: www.asim.uni-wuppertal.de/datenbank/own-experiments/bottleneck/bottleneck-no-2.html. Accessed 02. September 2015. Project funded by the German Science Foundation (DFG) under DFG-Grant No. KL 1873/1-1 and SE 1789/1-1.

van den Berg, J., Lin, M., and Manocha, D. (2008). Reciprocal velocity obstacles for real-time multi-agent navigation. In *IEEE International Conference on Robotics and Automation, 2008 (ICRA 2008)*, pages 1928–1935.

Vanillase (2011). Asimo at a honda factory. Wikimedia Commons. commons.wikimedia.org/wiki/File:ASIMO_4.28.11.jpg. Accessed 18. August 2014.

Varas, A., Cornejo, M. D., Mainemer, D., Toledo, B., Rogan, J., noz, V. M., and Valdivia, J. A. (2007). Cellular automaton model for evacuation process with obstacles. *Physica A: Statistical Mechanics and its Applications*, 382(2):631–642.

Vermeulen, A., Ambler, S. W., Bumgardner, G., Metz, E., Misfeldt, T., Shur, J., and Thompson, P. (2000). *The Elements of Java Style*. Cambridge University Press, Cambridge.

Vesely, F. J. (2001). *Computational Physics: An Introduction*. Springer, New York, 2nd edition.

Vicsek, T. and Zafeiris, A. (2012). Collective motion. *Physics Reports*, 517(3–4):71–140.

von Neumann, J. (1963). The general and logical theory of automata. In Taub, A. H., editor, *Collected Works: Design of Computers, Theory of Automata and Numerical Analysis*, volume 5. Pergamon Press.

von Neumann, J. (1966). *Theory of Self-Reproducing Automata*. University of Illinois Press.

von Sivers, I. (2013). Numerische Methoden zur Optimierung der Schrittrichtung und -weite in einem Modell der Personenstromsimulation. Master's thesis, Fernuniversität in Hagen.

von Sivers, I. and Köster, G. (2015). Dynamic stride length adaptation according to utility and personal space. *Transportation Research Part B: Methodological*, 74:104 – 117.

von Sivers, I., Seitz, M. J., and Köster, G. (2015). How do people search: a modelling perspective. In *11th International Conference on Parallel Processing and Applied Mathematics*.

von Sivers, I., Templeton, A., Köster, G., Drury, J., and Philippides, A. (2014). Humans do not always act selfishly: Social identity and helping in emergency evacuation simulation. In *The Conference in Pedestrian and Evacuation Dynamics 2014*, Transportation Research Procedia, pages 585–593, Delft, The Netherlands.

von Sivers, I., Templeton, A., Künzner, F., Köster, G., Drury, J., Philippides, A., Neckel, T., and Bungartz, H.-J. (2016). Modelling social identification and helping in evacuation simulation. *arXiv*, 1602.00805(v1).

Vuchic, V. R. (1999). *Transportation for Livable Cities.* Rutgers Center for Urban Policy Research.

Vuchic, V. R. (2005). *Urban Transit: Operations, Planning and Economics.* John Wiley & Sons.

Wahle, J., Neubert, L., Esser, J., and Schreckenberg, M. (2001). A cellular automaton traffic flow model for online simulation of traffic. *Parallel Computing*, 27(5):719–735.

Warren, P. A. and Rushton, S. K. (2009). Optic flow processing for the assessment of object movement during ego movement. *Current Biology*, 19(18):1555–1560.

Was, J., Gudowski, B., and Matuszyk, P. (2006). Social distances model of pedestrian dynamics. In El Yacoubi, S., Chopard, B., and Bandini, S., editors, *Cellular Automata*, volume 4173 of *Lecture Notes in Computer Science*, pages 492–501. Springer Berlin Heidelberg.

Was, J. and Lubas, R. (2013). Adapting social distances model for mass evacuation simulation. *Journal of Cellular Automata*, 8:395–405. Journal of Cellular Automata, Old City Publishing.

Was, J. and Lubas, R. (2014). Towards realistic and effective agent-based models of crowd dynamics. *Neurocomputing*, 146:199–209.

Weidmann, U. (1992). *Transporttechnik der Fussgänger*, volume 90 of *Schriftenreihe des IVT*. Institut für Verkehrsplanung, Transporttechnik, Strassen- und Eisenbahnbau (IVT) ETH, Zürich, 2nd edition.

Weifeng, F., Lizhong, Y., and Weicheng, F. (2003). Simulation of bi-direction pedestrian movement using a cellular automata model. *Physica A: Statistical Mechanics and its Applications*, 321(3–4):633–640.

Wellek, S. (2010). *Testing Statistical Hypotheses of Equivalence and Noninferiority.* Chapman & Hall/CRC, 2nd edition.

Westervelt, E. R., Grizzle, J. W., Chevallereau, C., Choi, J. H., and Morris, B. (2007). *Feedback Control of Dynamic Bipedal Robot Locomotion.* Control and Automation. Taylor & Francis / CRC.

Whitham, G. B. (1990). Exact solutions for a discrete system arising in traffic flow. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 428(1874):49–69.

Wiener, N. (1961). *Cybernetics, or Control and communication in the Animal and the Machine.* MIT Press, Cambridge, MA, 2nd edition.

Wijermans, N. (2011). *Understanding Crowd Behaviour: Simulating Situated Individuals*. PhD thesis, Rijksuniversiteit Groningen.

Wilson, M. (2002). Six views of embodied cognition. *Psychonomic Bulletin & Review*, 9(4):625–636.

Winter, D. A. (2009). *Biomechanics and Motor Control of Human Movement*. John Wiley & Sons, Hoboken, NJ, 4th edition.

Wolfram, S. (1983). Statistical mechanics of cellular automata. *Review of Modern Physics*, 55:601–644.

Wolfram, S. (1984). Cellular automata as models of complexity. *Nature*, 311:419–424.

Wölki, M., Schadschneider, A., and Schreckenberg, M. (2006). Asymmetric exclusion processes with shuffled dynamics. *Journal of Physics A: Mathematical and General*, 39(1):33.

Xu, Q., Mao, B., Liang, X., and Feng, J. (2012). Simple cognitive heuristics applied to modeling pedestrian behavior dynamics. In *8th International Conference on Traffic and Transportation Studies (ICTTS 2012)*, volume 43 of *Procedia - Social and Behavioral Sciences*, pages 571–578.

Yamazaki, N., Ishida, H., Kimura, T., and Okada, M. (1979). Biomechanical analysis of primate bipedal walking by computer simulation. *Journal of Human Evolution*, 8(3):337–349.

Yu, W. J., Chen, R., Dong, L. Y., and Dai, S. Q. (2005). Centrifugal force model for pedestrian dynamics. *Physical Review E*, 72:026112.

Zanlungo, F., Ikeda, T., and Kanda, T. (2014). Potential for the dynamics of pedestrians in a socially interacting group. *Physical Review E*, 89:012811.

Zarboutis, N. and Marmaras, N. (2004). Searching efficient plans for emergency rescue through simulation: the case of a metro fire. *Cognition, Technology & Work*, 6(2):117–126.

Zhang, P., Jian, X.-X., Wong, S. C., and Choi, K. (2012). Potential field cellular automata model for pedestrian flow. *Physical Review E*, 85(2-1):021119.

Zhang, Q. and Han, B. (2011). Simulation model of pedestrian interactive behavior. *Physica A: Statistical Mechanics and its Applications*, 390(4):636–646.

Zheng, X., Zhong, T., and Liu, M. (2009). Modeling crowd evacuation of a building based on seven methodological approaches. *Building and Environment*, 44(3):437–445.

Zhu, Z. C., Sui, Z., Tian, Y. T., and Jiang, H. (2014). Modeling and control of passive dynamic walking robot with humanoid gait. *Applied Mechanics and Materials*, 461:903–907.

Ziemer, C., Plumert, J., Cremer, J., and Kearney, J. (2009). Estimating distance in real and virtual environments: Does order make a difference? *Attention, Perception, & Psychophysics*, 71(5):1095–1106.

Zönnchen, B. (2013). Navigation around pedestrian groups and queueing using a dynamic adaption of traveling. Bachelor's thesis, University of Applied Sciences Munich.