FACULTY OF INFORMATION TECHNOLOGY CTU IN PRAGUE

Assignment of bachelor's thesis

Title:	Simulation of recreational pedestrian movement along the trail in the Luzen valley
Student:	Tomáš Novotný
Supervisor:	doc. Ing. Pavel Hrabák, Ph.D.
Study program:	Informatics
Branch / specialization:	Artificial Intelligence 2021
Department:	Department of Applied Mathematics
Validity:	

Instructions

Opening a trail through Luzen valley is a highly discussed topic in NP Šumava. The thesis topic is motivated by an idea of responsible authorities in NP to simulate various options of restricted access to this valley. The thesis itself should serve as a pilot project opening possible collaboration with NP Šumava regarding this topic.

The goal of the thesis is to introduce an agent based simulation combining three mathematical models:
A) Non-homogeneous random arrival process at both ends of the trail.
B) Fundamental-diagram (FD) based model transporting pedestrian agents to the bottleneck represented by a wooden walkway.
C) Microscopic model of the wooden walkway.
The main focus of the Bachelor thesis should be given on model B). Models A) and C) may be elaborated in subsequent theses.

Specific points of the thesis assignment are:
1. Make a survey of models and empirical findings related to non-emergent recreational pedestrian movement.
2. Study basic principles of agent-based modelling of pedestrian dynamics focusing on FD and microscopic models. Study basic principles of random arrival processes.
3. Suggest and implement an agent-based model combining three mathematical models mentioned above. Focus on FD part of the model implementing the density estimates from (Vacková and Bukáček, 2023).

Electronically approved by Ing. Magda Friedjungová, Ph.D. on 30 November 2023 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

4. Suggest, which empirical findings can be used for parameter calibration of the proposed model.
5. Perform simulation experiments and compare the results with the empirical findings from point 1 not used for calibration.

Literature:
Jana Vacková & Marek Bukáček (2023) Kernel estimates as general concept for the measuring of pedestrian density, Transportmetrica A: Transport Science, DOI: 10.1080/23249935.2023.2236236

Electronically approved by Ing. Magda Friedjungová, Ph.D. on 30 November 2023 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Bachelor's thesis

SIMULATION OF RECREATIONAL PEDESTRIAN MOVEMENT ALONG THE TRAIL IN THE LUZEN VALLEY

Tomáš Novotný

Faculty of Information Technology
Department of Applied Mathematics
Supervisor: doc. Ing. Pavel Hrabák, Ph.D.
May 13, 2024

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis: Novotný Tomáš. *Simulation of recreational pedestrian movement along the trail in the Luzen valley.* Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2024.

Contents

List of Figures

List of Tables

I would like to express sincere gratitude to my supervisor, doc. Ing. Pavel Hrabák, Ph.D., for actively pushing me in the right direction and guiding me through all parts of this thesis. I would also like to thank Ing. Jana Vacková, for continuously providing me with new thoughts and ideas. And last but not least, my family and friends for expressing appreciation, which was a great motivation.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular the fact that the Czech Technical University in Prague has the right to conclude a licence agreement on the utilization of this thesis as a school work pursuant of Section 60 (1) of the Act.

In Prague on May 13, 2024

Abstract

This bachelor thesis presents a stochastic agent-based model that simulates pedestrian movement in the Luzen Valley, National Park Šumava. The model merges three distinct mathematical models. Random arrivals of agents are modeled by means of a Poisson process. Further, a fundamental diagram-based model is introduced to transport agents to the estimated bottleneck of the trail. The third microscopical model represents a movement of pedestrians on the bottleneck. The transport section introduces the agent mass in the perceived surroundings of the agent as a novel concept to model interaction between agents that represent pedestrians. Furthermore, kernel estimates are implemented as an individual mass generated by each agent. Due to the focus of this thesis on the transport section of the model, the bottleneck section is represented by a very simple cellular automaton, a totally asymmetric simple exclusion process.

Further, this thesis successfully implemented the model in a simulation tool that has been developed to conduct simulation experiments with the model. The model parameters were calibrated using empirical findings from pedestrian movement. Correctness of kernel implementation was confirmed by one of the experiments. Furthermore, the experiments with the arrival intensity revealed that jams occur at the average arrival intensity of 48 pedestrians per minute. This value corroborates the reasonableness of the transport section model, as it aligns with the empirical findings that were not used for calibration. Additionally, the experiment results validated the assumption that heterogeneity in speed of agents and heterogeneity in arrival intensity leads to earlier stoppages.

Keywords pedestrian trail model, recreational pedestrian movement, stochastic agent based model, trail capacity, agent mass, fundamental diagram, kernel density estimates, random pedestrian arrivals, bottleneck flow

Abstrakt

Tato práce představuje stochastický agentní model, který simuluje pohyb chodců v Luzenském údolí v Národním parku Šumava. Model spojuje tři odlišné matematické modely. Náhodné příchody agentů jsou modelovány pomocí Poissonova procesu. Dále je zaveden model založený na fundamentálním diagramu pro dopravu agentů k odhadovanému úzkému hrdlu stezky. Třetí mikroskopický model reprezentuje pohyb chodců v úzkém hrdle. Dopravní model zavádí hmotu agentů ve vnímaném okolí agenta jako nový koncept pro modelování interakce mezi agenty reprezentující chodce. Jádrové odhady jsou navíc zavedeny jako individuální hmota generovaná každým agentem. Vzhledem k zaměření této práce na dopravní úsek modelu je úzké hrdlo stezky reprezentováno velmi jednoduchým celulárním automatem.

V praktické části této práci je model úspěšně implementován do simulačního nástroje, který byl vyvinut za účelem provádění simulačních experimentů s modelem. Parametry modelu byly kalibrovány na základě empirických pozorování o pohybu chodců. Správnost implementace jádrových odhadů byla potvrzena jedním z experimentů. Experimenty s intenzitou příchodu dále ukázaly, že k zácpám dochází při intenzitě 48 chodců za minutu. Tato hodnota potvrzuje smysluplnost modelu dopravního úseku, protože se shoduje s empirickými pozorováními, které nebyly použity pro kalibraci. Kromě toho výsledky experimentů potvrdily předpoklad, že různorodost v rychlosti chodců a různorodost v intenzitě příchodů vede k dřívějším zácpám.

Klíčová slova model pěší stezky, rekreační pohyb chodců, stochastický agentní model, kapacita stezky, hmota agenta, fundamentální diagram, jádrové odhady hustoty, náhodné příchody chodců, úzké hrdlo stezky

List of Abbreviations

ABM	Agent Based Model
CA	Cellular Automata
FD	Fundamental Diagram
TASEP	Totally Asymmetric Simple Exclusion Process
ASEP	Asymmetric Simple Exclusion Process
NaSch	Nagel-Schreckenberg model
i.i.d.	Independent and Identically Distributed
LOS	Level of Aervice
ODE	Ordinary Differential Equation

Introduction

The Luzen Valley is currently a strictly protected and inaccessible area within the first zone of the Šumava National Park. It houses bogs and rare plant varieties. In addition, it represents a survival spot for the ruffed grouse, a critically endangered bird that rarely thrives in Europe.

The accessibility of this valley is for many years under extensive debate. Simulating various options for restricted access could provide valuable insights for the current debate surrounding the potential opening of Luzen Valley to the public. The purpose of the thesis is to establish cooperation with the national park administration to estimate the capacity and select a suitable restriction model for pedestrian movement along the trail.

Pedestrian movement, also known as pedestrian dynamics, is a research field within transportation systems. The theory of pedestrian dynamics is historically based on research conducted on vehicle traffic, which was the first transportation system to be studied.

Pedestrian traffic is a much more complex issue than vehicle traffic. It is considered to be the most advanced of all transportation systems among biological systems. For instance, pedestrian traffic is two-dimensional in nature and pedestrian reactions cannot be described solely by the distance from surrounding pedestrians. In addition, pedestrian flow is influenced by cultural differences, elevation of the terrain, and weather.

Pedestrian flow involves several unique phenomena that often differ from modeling highway traffic. Section 3.1 discusses selected observation and empirical findings related to the pedestrian movement in detail. It reveals that the capacity of a bottleneck is a crucial factor in modeling pedestrian flow. This point is not only where walking speed decreases, but it can also lead to jamming phenomena.

Modeling pedestrian flow has become a popular research topic in recent decades. The relevance of this topic is increasing in light of population growth and urbanization tendencies across communities. The goal is to build safe and effective transportation environment.

Three principal categories of research methods are employed in the field of pedestrian transportation systems, namely laboratory experiments, empirical methods, and numerical simulations. Conducting experiments under laboratory conditions represents a significant source of pedestrian behavior data. However, certain experiments are either impractical or unsafe, and the results may be distorted due to the participants' awareness of the experiment. In contrast, the empirical method collects real-world data and is therefore more reliable. As tracking technology continues to evolve, there is a growing concern about the potential for infringements on privacy and the limitations of collecting data on streets.

Significant improvements in computational performance have opened up new possibilities for performing numerical simulations of pedestrian movement. There are two main approaches to model pedestrian movement, namely knowledge-based models that mimic the pedestrian movement mechanism in explicit ways and data-driven models that use statistical methods to infer pedestrian behavior. In recent years, research has focused on the data-driven approach, with the

advent of machine learning. Initially, optimization techniques were integrated into knowledge-based models with the objective of reducing the discrepancy between simulation and reference data. More recently, pedestrian movement simulation has been approached as a time sequence generation problem and solved using neural networks.

Compared to vehicle traffic modeling, there exist less empirical data for pedestrian flow. Recently, the analysis of video footage has emerged as the most effective data source for pedestrian modeling. Several open-source datasets have been created that can be used as training datasets for the machine learning methods.

Nevertheless, the potential for simulating recreational pedestrian movement has yet to be fully explored. The state of the art models of pedestrian movement and the models that represent recreational pedestrian movement are presented in Section 3.3. However, the Luzen valley trail requires a custom model.

The primary goal of the thesis is to introduce and implement a stochastic agent-based model that represents recreational pedestrian movement in the Luzen valley. Agent-based modeling is the subject of Section 1.1. The introduced model consists of three distinct mathematical models, described in Chapter 4.

The first model deals with random arrivals of agents to the trail by means of a Poisson process, which is subject of Section 1.3. The second model addresses transportation of agents from the edges of the trail to the bottleneck section. Crucial quantity in pedestrian dynamics is the fundamental diagram, which is introduced in Section 2.1. This section employs a fundamental diagram-based model and introduces a novel concept of interaction between agents, namely the agent mass in the perceived surroundings of the agent. Furthermore, kernel estimates, which are introduced in the article [1] and discussed in Section 2.2.2, are implemented as an individual mass generated by each agent. The third model addresses pedestrian movement on the bottleneck section of the trail. Due to the focus of this thesis on the transport section of the model, the bottleneck section is represented by a very simple cellular automaton, which is subject of Section 1.2.1.

Further goal of this thesis is to implement the agent-based model in a simulation tool to conduct experiments with the model and log the trajectories of agents from simulations. Chapter 5 deals with the implementation of the tool in detail. The secondary goal is to create a script providing a basic animation of the trajectories from the simulation.

To ensure the credibility of the model, the thesis goal is to calibrate the model parameters using empirical findings on pedestrian movement, which are presented in Section 3.1 and Section 3.2. The parameter calibration is presented in Section 5.1.

The last objective of the thesis is to evaluate a capability of the model to emulate recreational pedestrian movement. For this purpose, various experiments were conducted in Section 6.2, and the results were compared with the values presented in Chapter 3.

This thesis contributes to the academic community by introducing an agent-based model of recreational pedestrian movement. Furthermore, the thesis introduces a novel concept for modeling interactions between pedestrians. In addition, the thesis practically applies kernel density estimates presented in the article [1].

Chapter 1

Methodologies for Modeling Pedestrian Movement

There are various approaches to modeling pedestrian movement, interactions between pedestrians, and arrivals to the trail. In this chapter, the theory of selected modeling methods is formally introduced and contextualized within the model presented in Chapter 4.

Initially, the concept of a multi-agent system is introduced through definitions of basic theory, classification of multi-agent systems, agent types, and environmental characteristics. Further section introduces Cellular Automata, a class of models that represent transportation systems, and introduces its main representatives. Additionally, a random arrival process is formally established, and the statistical methods used in the thesis to simulate random arrivals are discussed. Finally, ordinary differential equation and equations of motion are introduced as a means of expressing pedestrian movement.

1.1 Multi-agent system

This section is based on the overview book [2], which discusses the concepts of artificial intelligence in detail, including intelligent agents. Initially, the definition of agent, agent function, and multi-agent systems is introduced. Secondly, the characteristics and types of agents are presented. Finally, task environment is defined and environment characteristics are presented. The theory is set in the context of the model introduced later in Section 4.2.

The following definitions formally describe essential concepts of multi-agent systems. Figure 1.1 contributes to the definitions by providing a schema of the interaction between the agent and the environment.

▶ **Definiton 1.1** (Agent)**.** *Anything that can perceive surroundings using its sensors and acts using actuators.*

▶ **Definiton 1.2** (Agent function)**.** *Maps perceived sequences to action sequences and is run by* ***agent program****.*

▶ **Definiton 1.3** (Multi-agent system)**.** *Collection of independent agents in shared environment.*

Agents can vary considerably depending on the application. **Robotic agents** exist in the physical world and have physical actuators. On the contrary, **software agents** exist in a virtual environment and work with data. The model introduced in this thesis uses software agents.

Despite the existence of differences in usage, agents share fundamental characteristics. These characteristics are listed below in an itemized list, each contextualized in the brackets within the agents introduced in this thesis.

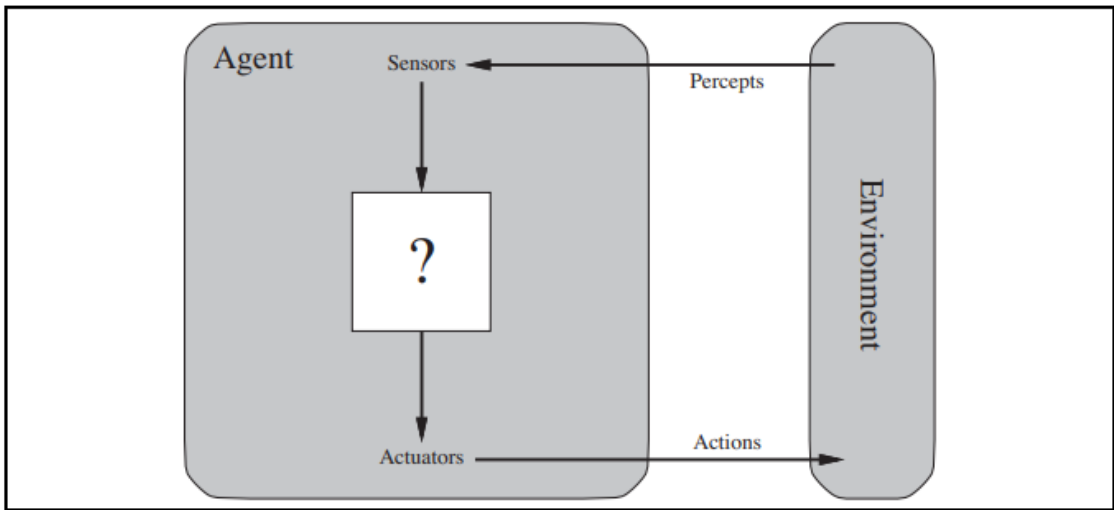

Figure 1.1 Schema of agent and environment interaction, taken from [2]

- being autonomous (agents have control over themselves),
- acting proactively to meet the objective (agents move towards the goal),
- making only purposeful activities (agents do not move backwards),
- interacting with the environment without having full control over it (agents adjust speed based on the surrounding environment),
- coordination, cooperation, negotiation with other agents (relations between agents are achieved through interaction).

Since the definition of the agent is too general, there are other ways of categorising agents into types. Based on their relationship to other agents, the following types of agents can be distinguished,

1. isolated,
2. cooperative,
3. self-interested,
4. combination of the above.

The agents introduced in this thesis are classified as a combination of cooperative and self-interested agents. To be complete, isolated agents that ignore other agents were used in some of the simulation experiments presented in Chapter 6.

One way to distinguish between types of agents is by their behaviour. There are two basic types, **simple reflex agents** and **model based agents**. Both types of agents employ if-then rules, yet model-based agents possess an internal state that extends beyond that of simple reflex agents. It is noteworthy that the agents introduced in the thesis are of the simple reflex agent type.

Further, there are **goal-based agents** and **utility-based agents**. Goal-based agents accept goals and plan actions based on those goals. Utility-based agents, on the other hand, use a utility function that maps the state of the environment to a real number representing the quality of the state, and behave based on that value.

In addition, **rationality** is a concept for making events that maximize performance. **Rational agents** are useful in applications in which tasks are accomplished. This type of agents is not a suitable approach to model pedestrian movement, in which the objective is to achieve authenticity and *uncertainty*.

The agent's behavior and success rate is strongly dependant on the complexity of the environment. Therefore, the agent design is intertwined with the specified environment. **Task Environments** (PEAS) specifies the multi-agent problem whose solution is the design of a suitable agent. The specification of the problem consists of the following four tasks, which are supplemented in the brackets by the context of the environment introduced in the thesis, namely

1. performance measure (no performance measure needed),
2. environment (agent mass profile, position on the trail),
3. actuators (generating an individual agent mass),
4. sensors (perceived surroundings of the agent).

As with agents, there is a wide variety of agent environments. The environment in the mentioned model is

- partially observable (due to perception of the surrounding density profile),
- deterministic (based on the current state, the next state is determined),
- sequential (next episode depends on the actions taken in the previous episode due to changing position),
- static (environment cannot change in one time instant while the agent is adjusting speed),
- continuous (both time and position).

1.2 Cellular Automata

Cellular automata (CA) belong to the computationally efficient simple programs that emulate processes and particle systems and are used to solve interdisciplinary problems, such as pedestrian movement. Initially, the model is defined and fundamental characteristics are presented. Secondly, the use-cases and extension possibilities are discussed. Further, selected representatives are introduced, and the possibilities of using these models in the context of pedestrian movement modeling are discussed. The most basic representative, totally asymmetric simple exclusion process (TASEP), used as the model introduced in Section 4.3, is discussed in detail.

The following paragraphs are based on the overview book [3], which discusses simple programs and models used to simulate processes in detail, including CA and its variants.

The linear variant of CA consists of a structure of cells, called a lattice, each of which has a state, typically from finite set. There is a definite rule that determines the state of a given cell based on the state of the cells in the neighborhood at each time step. The program behavior and rules have a visual representation. Figure 1.3 shows behavior visualization of simple CA model rules.

For the purpose of modeling bidirectional pedestrian movement, a **two-dimensional variant** of the CA is needed. Although models with two-dimensions share similar behavior with models with one-dimension, two-dimensional CA show less regular patterns in step visualization.

Simple and proper rules are required to produce highly complex behavior. Even when all cells follow exactly the same rule, experiments show that different configurations with different sequences produce all sorts of different kinds of behavior. A popular example is *Game of Life*

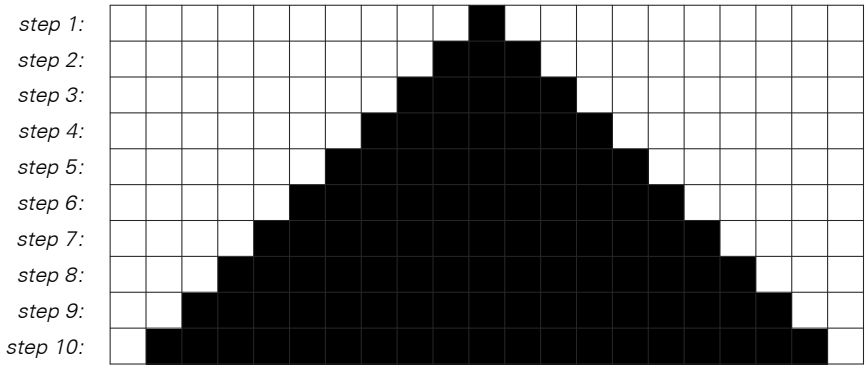

Figure 1.2 Visualization of 10 steps of simple CA model, taken from [3]

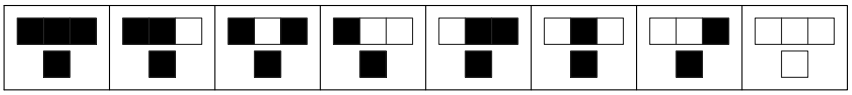

Figure 1.3 Visualization of rule examples, taken from [3]

introduced in the article [4]. To reproduce detailed natural system, such as pedestrian movement, more complicated rules may be convenient although they do not bring fundamentally new features.

There exist several extensions of CA. For example, **mobile automata** differs in that it updates only one cell at a time step and not all in parallel. Next, **continuous extension** generalizes the default definition using a continuous range of possible states.

Cellular agent systems are computational models that combine the principles of CA and agent-based modeling. They are used to simulate complex systems comprising individual agents that interact with one another within a grid-like structure. This concept is employed in the model presented further in Section 4.3, as each agent possesses unique properties, such as velocity and direction.

In a cellular agent system, each cell is either occupied by an agent or not. Each agent has its own state, behavior, and rules. Furthermore, agents can move, interact with surrounding agents, and update their states based on local rules and environmental conditions.

A useful resource for the following paragraphs is the overview book [5], which provides a detailed analysis of transportation systems.

CA models are a popular approach in modeling pedestrian movement. Space discretion in these models can be derived from the assumption that each pedestrian occupies a similar area.

The most basic model of all cellular models is the *Nagel-Schreckenberg* (NaSch) model, in which both time and space are discrete. A special case of this model is the TASEP with all its modifications. It is noteworthy that these models are primarily utilized in vehicle traffic models and in single-file dynamics systems.

The article [6] introduced more sophisticated model that has become popular in pedestrian flow dynamic, which is called **floor field**. The book [5] states that the model is capable of comprehending the collective effects observed in pedestrian dynamics. It is based on the idea that each direction has a different probability of being selected and is inspired by *chemostaxis* phenomena used by insects.

There are also models that use non-CA approaches. One commonly used in pedestrian dynamics is the **social-force** model, described in the article [7]. Behavioral changes are implemented using the concept of a social field that affects pedestrians, and the main contribution of force

comes from the territorial effect.

1.2.1 Totally Asymmetric Exclusion Process

The article [8] states that models of pedestrian traffic are strongly related to the TASEP because it is considered to be the simplest possible stochastic transport model. Furthermore, it is generally regarded as a generic model for many particle systems. In addition, the paper provides the basis for the following paragraphs.

The model is based on the asymmetric simple exclusion process (ASEP) that consists of one-dimensional finite lattice and boundary points, called bounds, typically distinguished into left and right. Each cell is in one of two states, namely unoccupied or occupied. In comparison to ASEP, there is only one moving direction. In comparison to the NaSch model, this model allows only two different speed levels.

All particles are moving in the same direction, typically from left to right, as depicted in Figure 1.4. During the particle update, each particle either moves to the subsequent cell if it is unoccupied or remains at the same cell. The update frequency for each particle is determined by the update probability.

According to the update strategy, **parallel** and **sequential updates** are distinguished, whereby parallel updates are more common in modeling traffic flow. Parallel updates performs injection, hopping, and removal simultaneously to all sites. Furthermore, important update strategies, namely random-sequential, and time-continuous updates, are introduced in the following paragraphs. The source of the following text is the article [9], which studies different types of updates for ASEP in detail.

Random-sequential update picks one of the particles, each with the same probability. If the subsequent cell is empty, the particle moves with probability p. In general, the particle on the boundary of the lattice is removed with probability β, and new particle moving to the lattice with different probability α, as shown in Figure 1.4.

The model introduced in Section 4.3, in which each particle represents an agent with individual optimal velocity, implements a time-continuous update strategy that shares stationary properties with random-sequential update. In contrast, it generates a time period, in which the particle waits for the subsequent update. During update, the particle hops to the subsequent cell if the cell is empty or waits for the next update.

Further, the book [5] distinguishes two main classes of TASEP models according to boundary conditions, i.e. behavior of particles on the edges of the one-dimensional lattice. With **periodic boundary** conditions, the lattice forms close circuit where particles can hop from the end of the lattice to the beginning. The appropriate boundary condition for pedestrian traffic is the **open boundary**, in which new particles emerge on the left boundary of the lattice and exiting particles leave the lattice on the right boundary.

The open boundary is realized by infinite reservoirs at the edges of the lattice. This idea is illustrated in Figure 1.4. On the left boundary, the particles are inserted at a rate α and on the right boundary, the particles are removed with the rate β. To be complete, each particle has the same update probability p.

1.3 Random Arrival Process

In this thesis, random arrival process results in a sequence of times of events. An important process that simulates random arrivals is the Poisson process. Initially, Exponential distribution, Poisson distribution and Poisson process are defined. Secondly, definition of the Poisson process and definition of non-homogeneous Poisson process is introduced. Finally, procedure to retrieve a trajectory of random arrival process is presented. The source for this section are the lecture notes [10], which are focused on application of selected statistical methods.

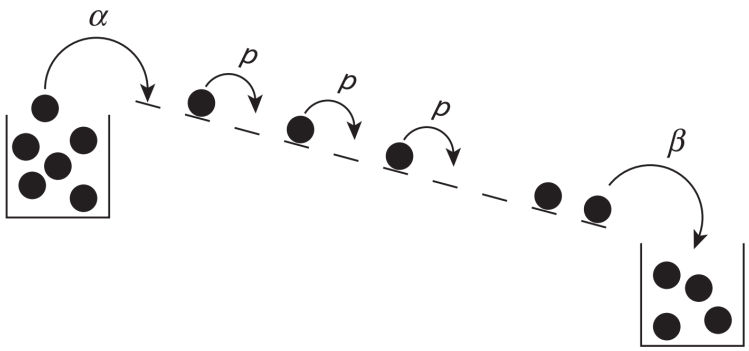

Figure 1.4 Illustration of TASEP with open boundary, taken from [5]

The following definitions of probability distributions are necessary to define the Poisson process.

▶ **Definiton 1.4** (Exponential distribution)**.** *Continuous random variable X has Exponential distribution with parameter λ $\overset{def}{\iff}$ the density function $f_X(x) = \lambda \cdot e^{-\lambda x}$, denoted as $X \sim Exp(\lambda)$.*

▶ **Definiton 1.5** (Poisson distribution)**.** *Discrete random variable X has Poisson distribution with parameter λ $\overset{def}{\iff}$ $P(X = k) = \frac{\lambda^k}{k!} e^{-k}, k \in \mathbb{N}_0$, denoted as $X \sim Poisson(\lambda)$.*

Poisson process is an example of a counting process that is established in the following definition.

▶ **Definiton 1.6** (Counting process)**.** *Stochastic process $\{N_t \,|\, t \in [0, +\infty)\}$ is a counting process if it meets the following conditions,*

(i) $N_t \geq 0$,

(ii) $N_t \in \mathbb{Z}$,

(iii) $s \leq t \Rightarrow N_s \leq N_t$.

In the following definition is established the homogeneous Poisson process.

▶ **Definiton 1.7** (Poisson process)**.** *The process $\{N_t \,|\, t \in [0, +\infty)\}$ is a Poisson process if the following apply,*

(i) $N_0 = 0$ *almost surely,*

(ii) $N_t - N_s \sim Poisson(\lambda(t - s))$ *for all* $t > s \geq 0$,

(iii) $\{N_t\}$ *has independent increments, i.e., for all* $k \in \mathbb{N}$ *and for all* $0 \leq t_0 < t_1 < \cdots < t_k$, $N_{t_1} - N_{t_0}, N_{t_2} - N_{t_1}, \ldots, N_{t_k} - N_{t_{k-1}}$ *are independent.*

The homogeneous Poisson process has one other equivalent definition using a trajectory. Using this definition, the random arrivals presented in Section 4.1 are constructed.

▶ **Definiton 1.8** (Poisson process)**.** *Let $\{X_j \,|\, j \in \mathbb{N}\}$ be random variables i.i.d. (independent and identically distributed) random variables with distribution $Exp(\lambda)$.*
Let us define trajectory $\{T_n \,|\, n \in \mathbb{N}\}$ as follows,

$$T_0 = 0, \quad T_n = T_{n-1} + X_n = \sum_{j=1}^{n} X_j, \quad n \in \mathbb{N}\,.$$

Then the random process $\{N_t \,|\, t \in [0, +\infty)\}$*, where*

$$N_t(\omega) := \max\{n \in \mathbb{N}_0 \,|\, T_n(\omega) \leq t\}\,,$$

is called a Poisson process.

A non-homogeneous Poisson process, which is established in the following definition, is employed to capture the variable intensity of arrivals over time.

▶ **Definiton 1.9** (Non-homogeneous Poisson process)**.** *Let* $\lambda(r)$ *be a integrable function on a finite interval.*

Then the process $\{N_t \,|\, t \geq 0\}$ *is called an non-homogeneous Poisson process with intensity* $\lambda(r)$ *if the following apply,*

(i) $N_0 = 0$ *almost surely,*

(ii) $\{N_t\}$ *has independent increments,*

(iii) $N_t - N_s \sim Poisson\left(\int_s^t \lambda(r)\, dr\right)$ *for* $t > s$*.*

It is possible to construct the non-homogeneous Poisson process with intensity $\lambda(t)$ using following theorem. The idea is depicted in Figure 1.5.

▶ **Theorem 1.10** (Thinning)**.** *Let us consider a Poisson process* $\{N_t | t \geq 0\}$ *with intensity* λ*. Let* $p(t)$ *be a function integrable on a finite interval. Further, at the event time* t *happens an event of type* 1 *with probability* $p(t)$*, and an event of type* 2 *with probability* $1 - p(t)$*. Let us define processes* $\{N_t^{(1)} | t \geq 0\}$*, and* $\{N_t^{(2)} | t \geq 0\}$ *that record the event count of type* 1*, type* 2 *respectively. Then the process* $\{N_t^{(1)}\}$ *counting events of type* 1 *is an inhomogeneous Poisson process with intensity* $\lambda(t) = \Lambda \cdot p(t)$*.*

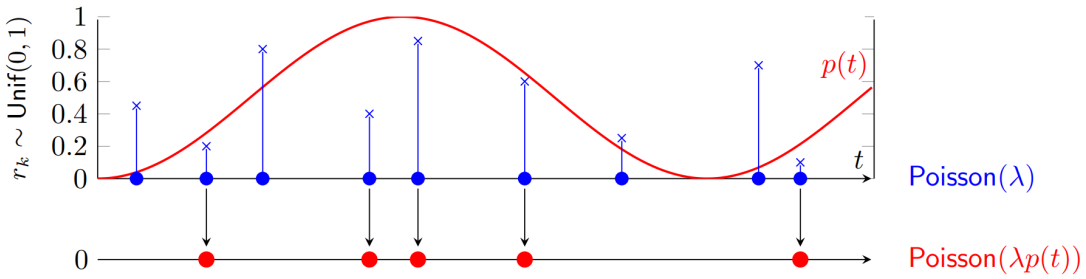

Figure 1.5 Illustration of non-homogeneous thinning, taken from [10]

Let us define Λ as the maximum intensity and $p(t)$ as the function that determines the variable probability of the event at time t, i.e.

$$\Lambda = \max\{\lambda(t) \,|\, t \geq 0\}\,, \tag{1.1}$$

$$p(t) := \frac{\lambda(t)}{\Lambda}\,. \tag{1.2}$$

The process accepting events with probability $p(t)$ is a non-homogeneous Poisson process with intensity $\Lambda p(t) = \lambda(t)$.

The previous definitions and theorems can be used to construct the times of non-homogeneous random arrivals. Initially, the times of events $\{T_n \,|\, n \in \mathbb{N}\}$ of the homogeneous Poisson process with the intensity λ are generated using definition 1.8. To achieve non-homogeneous Poisson process, each event time t is accepted with the probability $p(t)$ from Equation (1.2).

1.4 Equations of Motion

The motivation for this section is the model introduced in Section 4.2, in which the velocity of each agent is described by the equation of motion presented in Equation (4.8). The valuable source for this section is the book [11] that introduces the fundamentals of ordinary differential equations.

To define the equation of motion, ordinary differential equations must be first introduced. In addition, a numerical method that approximates the solution is presented. Further, velocity and acceleration are defined. Finally, the approach in the model introduced in this thesis is briefly presented.

Ordinary differential equation (ODE) is an equation

$$y'(x) = f(y, x), \text{ where } y : \mathbb{R} \to \mathbb{R} \wedge x \in \mathbb{R}\,. \tag{1.3}$$

The equation can be extended with initial conditions. Initial conditions determine the function value in the point $a \in \mathbb{R}$, e.g.

$$y(0) = 0\,. \tag{1.4}$$

Differential equation that contains second-degree derivative is called **second-order differential equation**. This equation is represented by the second-degree derivative of the function $y(x)$ on the left side of the equation and a function of the derivative of the function $y(x)$, the function $y(x)$ and the variable x on the right side of the equation, i.e.

$$y''(x) = f(y', y, x)\,. \tag{1.5}$$

In the following, the **explicit Euler method** is introduced. It numerically solves the differential equation (1.3) with initial condition

$$y'(x_0) = C\,, \text{ where } x_0 \in \mathbb{R} \wedge C \in \mathbb{R}\,. \tag{1.6}$$

Initially, the domain of function y is discretized into sequence

$$(x_n)_{n=0}^{\infty}, \text{ where } x_n = x_0 + h \cdot n\,.$$

Further, the method approximates the function $y(x)$ in these discrete points with sequence $(y_n)_{n=0}^{\infty}$, i.e. $y_n \approx y(x_n)$. The method explores the points of sequence $(y_n)_{n=0}^{\infty}$ by replacing the expressions that contain the function $y(x)$ with values of sequence (y_n).

In the following example, the Euler method is used to numerically solve the following differential equation (1.7) with the initial condition $y(0) = 0$.

$$y'(x) = y^2(x) + x\,, \tag{1.7}$$

The derivative $y'(x)$ is replaced with expression $\frac{y_{n+1}-y_n}{x_{n+1}-x_n}$ and the indeterminate $y^2(x_n)$ is replaced with y_n^2, which leads to the following approximate solution,

$$y_{n+1} = y_n + h \cdot (y_n^2 + x_n)\,. \tag{1.8}$$

In order to use the explicit Euler method for the second-order differential equation, it is neccessary to transform the equation (1.5) to a system of coupled ODE of the first-order by substituting the first-order derivation $y'(x)$ by artificial variable v, i.e.

$$y' = v\,, \tag{1.9}$$

whereby applies that the derivation of this variable is equal to y'', i.e.

$$v' = f(v, y, x)\,. \tag{1.10}$$

In addition to the approximation of the target function $y(x)$ by $y_n \approx y(x_n)$, another variable, variable v from Equation (1.9), has to be approximated similarly to the variable y by a sequence $(v_n)_{n=0}^{\infty}$, where $v_n \approx v(x_n) = y'(x_n)$.

As in Equation (1.8), the approximate solution of the second-order differential equation is shown in the following equations,

$$y_{n+1} = y_n + h \cdot v_n \,, \tag{1.11}$$

$$v_{n+1} = v_n + h \cdot f(v_n, y_n, x_n) \,, \tag{1.12}$$

As mentioned, the equations of motion describe the motion as a function of time, which is in the following text denoted as variable $t \in [0; \infty]$. In context of Section 4.2, position of agent α at time t is denoted as $x_\alpha(t)$. Using time and position, velocity and acceleration are defined in the following definitions.

▶ **Definiton 1.11** (Velocity)**.** *Velocity is a change of position in a period of time, denoted as* $\dot{x}(t)$*.*

▶ **Definiton 1.12** (Acceleration)**.** *Acceleration is a change of velocity in a period of time, denoted as* $\ddot{x}(t)$*.*

Newtonian model of equations of motion describes acceleration $\ddot{x}$ at the time t by the force F that depends on a position x and velocity $\dot{x}$, i.e.

$$\ddot{x}(t) = \mathrm{F}(\dot{x}, x, t) \,.$$

This concept is taken over by the social-force model that determines a motion of each agent $\alpha \in \mathrm{A}$ at the time t by a force F, which is described in the article [7]. The force F depends on positions $\{x_\beta; \beta \in \mathrm{A}\}$ and velocities $\{\dot{x}_\beta; \beta \in \mathrm{A}\}$. Therefore, the trajectories are given by a coupled second-order differential equations,

$$\ddot{x}_\alpha(t) = \mathrm{F}(\{x_\beta; \beta \in \mathrm{A}\}, \{\dot{x}_\beta; \beta \in \mathrm{A}\}, t) \quad \text{for } \alpha \in \mathrm{A} \,. \tag{1.13}$$

In contrary, the model introduced in this thesis describes the motion of agents by a coupled first-order differential equations, presented in Equation (4.8). In order to achieve the greatest degree of simplicity, the velocity of each agent is solely dependent on the positions of all agents. Using the explicit Euler method, the approximate solution of this coupled equations is the following formula where parameter h denotes the parameter of discretization, i.e.

$$x_{\alpha,n+1} = x_{\alpha,n} + h \cdot \mathrm{V}_\alpha(\{x_{\beta,n} \,|\, \beta \in \mathrm{A}\}, h \cdot n) \,. \tag{1.14}$$

Chapter 2

Characteristics of Pedestrian Flow

This chapter presents selected concepts of pedestrian flow theory. It begins by introducing the fundamental diagram (FD) as a key characteristic of traffic and pedestrian flow, along with necessary quantities such as flow or density. The chapter then proceeds to discuss density estimate methods in pedestrian flow, including kernel estimates, as part of the model introduced in Section 4.2. Finally, this chapter introduces the level of service (LOS) as a measure of quality of service from a user's perspective.

2.1 Fundamental Diagram

Fundamental serves as a key characteristic of mutual interactions within a pedestrian group. There exists variety of basic parameters and functional forms characterizing FD. Initially, definitions of the characteristics are introduced. Furthermore, forms of fundamental diagram are presented and methods for its measuring are discussed.

Initially, key variables are introduced using definitions from the article [1]. The measurement of pedestrian flow is based on the hydrodynamic theory. The following two definitions introduce density, and flow.

▶ **Definiton 2.1** (Pedestrian Density)**.** *The pedestrian density ρ denotes the number of pedestrians per unit area. It is commonly measured as $\rho = \frac{N}{|A|}$, where N denotes the number of pedestrians and $|A|$ denotes the size of the area.*

▶ **Definiton 2.2** (Pedestrian Flow)**.** *The pedestrian flow J is a number of pedestrians crossing a checkpoint per unit time. It is commonly measured as $J = \frac{N}{\Delta T}$, where N denotes number of pedestrians crossed within time interval of the length ΔT.*

Using the definitions 2.1, 2.2 FD is formally established in the following definition.

▶ **Definiton 2.3** (Fundamental Diagram)**.** *The fundamental diagram is a relation between density and pedestrian flow, i.e.*

$$J = J(\rho)\,. \tag{2.1}$$

The book [5] states that the hydrodynamic relation $J = \rho v$ enables its three equivalent forms, namely $J(\rho)$, $v(\rho)$ and $v(J)$, shown in Figure 2.1. To be noted, model introduced in Section 4.2 is inspired by the form $v(\rho)$.

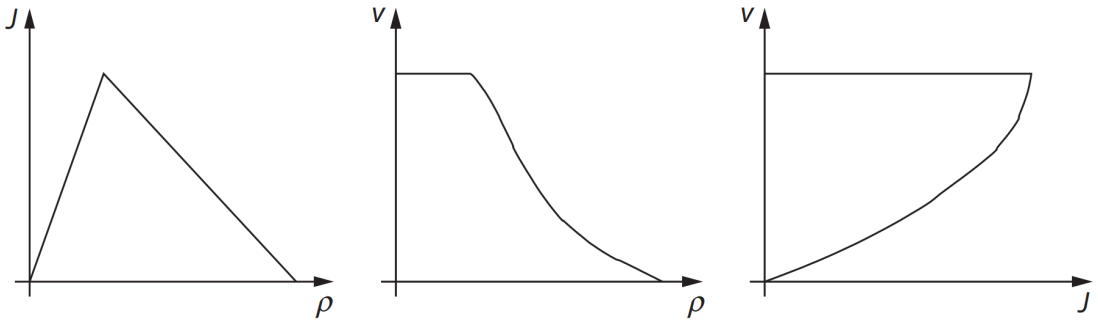

Figure 2.1 Generic shapes of fundamental diagram forms, taken from [5]

Moreover, the papers express concern that a single form may not be sufficient to characterize bidirectional traffic, due to differing perceptions of pedestrians in the opposite direction. These opinions diverge in the literature, and in general, the measurement of the fundamental diagram remains a topic of contention.

The book [5] further states that due to practical observations, the average speed of pedestrians in traffic remains independent of density at low densities. However, at some sufficiently high density, referred to as **critical density**, average speed decreases. At a particularly high density, referred to as **maximum density**, the speed reduces to the point of complete stopping.

One of the most simple forms is the *Greenshields model*, corresponding to the velocity-density relation

$$v(\rho) = v_{\text{opt}} \left(1 - \frac{\rho}{\rho_{\text{max}}}\right) , \tag{2.2}$$

shown on left in Figure 2.2.

More complex form the fundamental diagram, given by [12], has the analytical form

$$v(\rho) = v_{\text{opt}} \left[1 - \exp\left(-\rho_0 \left(\frac{1}{\rho} - \frac{1}{\rho_{\text{max}}}\right)\right)\right] . \tag{2.3}$$

and is shown on right in Figure 2.2.

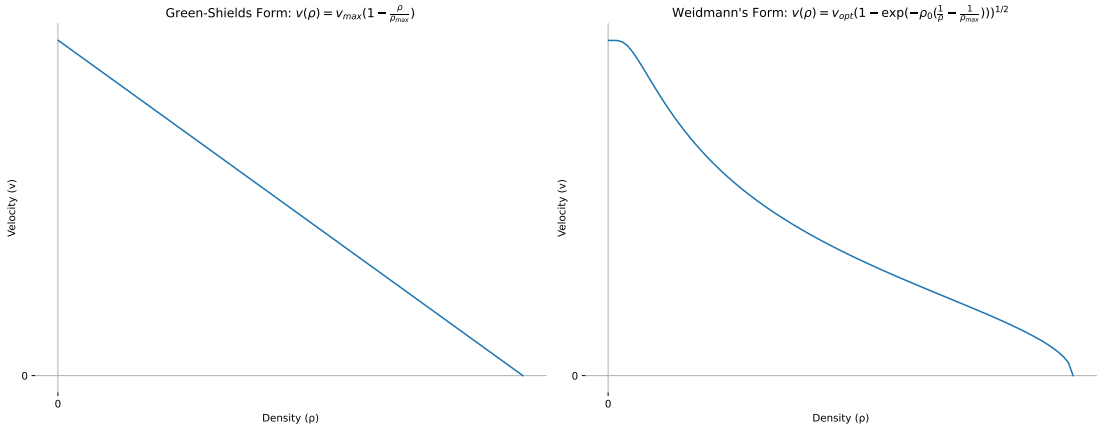

Figure 2.2 Velocity-density relation of Greenshields and Weidmann fundamental diagram

To be complete, the model introduced in Section 4.2 uses a modified Greenshields fundamental

diagram, given by the following formula and shown in Figure 4.6,

$$v(\rho) = \begin{cases} v_{\text{opt}} & \text{for } \rho < \rho_{\text{crit}}\,, \\ v_{\text{opt}} \left(1 - \frac{\rho - \rho_{\text{crit}}}{\rho_{\text{max}}}\right) & \text{for } \rho_{\text{crit}} < \rho < \rho_{\text{max}}\,, \\ 0 & \text{for } \rho_{\text{max}} < \rho\,, \end{cases} \tag{2.4}$$

Comparison of different fundamental diagrams can be found in the report [13]. Most of the experiment results show similarity to the function (2.4). It is noteworthy that Society of Fire Protection Engineers (SFPE) issues standards and guidelines that include recommendations how to conduct pedestrian flow studies.

2.2 Estimates of Pedestrian Density

Initially, the importance of pedestrian density is explained and the reasons for estimating pedestrian density are debated. Next, an overview of methods used to estimate pedestrian density is presented, and two popular methods are presented in detail.

As the article [1] discusses, the standard definitions of flow and density describe the global current state of the system. However, local estimates describe the traffic situation in more detail that is desired for the precise simulation of the pedestrian movement. Additionally, during density measurement using experimental data, the definition of pedestrian density produces scattered values. The methods of density estimates can approximate the continuous density.

The article [14] focuses on a comparison of different methods used to measure fundamental diagram. It considers the *X-T measure*, which is grid method, as the most effective measure. Another grid method is the *Voronoi diagram* measure, which is popular but computationally intensive space-based method presented further in this section.

The first discrete form of the grid method was introduced by [15]. This method divides the space into cells, with area its denoted as $|A|$, and each pedestrian contributes to the local density of the cell by value $\frac{1}{|A|}$. Continuous version of the grid method was introduced by [16] extending the individual contribution into $\frac{t/T}{|A|}$ including proportion of the time spent in the cell.

Later, smoother versions of density estimate methods were published. An overview of the methods can be found in the article [14]. Further follows an introduction of two selected methods, namely Voronoi diagram and kernel estimates. The source for the following text is the article [1] that analyzes kernel estimates of pedestrian density and compares them with other popular density estimation methods.

2.2.1 Voronoi Diagram

Voronoi diagram is one of the most popular and basic methods used to estimate density. The fundamental idea of this method is that each point $x \in \mathbb{R}$ is assigned to the nearest pedestrian α. This divides observed area A into subsets $A_\alpha \subset A$, referred to as **Voronoi cell**, for each pedestrian α. Resulting density at a point $x \in A_\alpha$ is then equal to $\frac{1}{A_\alpha}$.

Due to distribution oscillation in every timestep, the article [17] states that density estimate retrieved from Voronoi method is usually a time average taken over the time. Further characteristic of this method is a computational intensive. To be complete, the article [14] states that it is possible to consider a additional parameter as a size limit of the Voronoi cell. This prevents extending Voronoi cells into infinity when no boundaries of the diagram are set.

An illustration of the space division of Voronoi cells in two dimensions is shown in Figure 2.3.

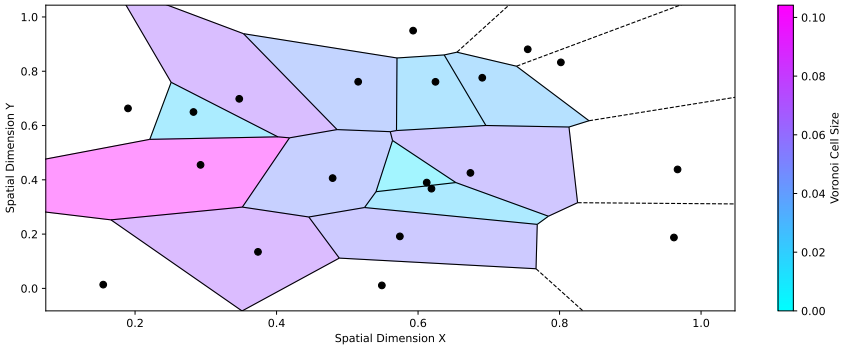

Figure 2.3 Voronoi diagram illustration, black points represent pedestrians

2.2.2 Kernel Estimates

Initially, notation of variables is defined and kernel estimates are derived from pedestrian density definition. Further, parameters are introduced, and interpretation of kernel estimates is discussed. Additionally, examples of kernels are presented and the use of a selected kernel in the model introduced in Section 4.2 is debated.

Kernel methods consider each pedestrian as a source of density distribution. Kernels are parameterised by kernel type and kernel size. Parameterization in the kernel method brings the desired features.

In the following text, the space will be limited to one dimension due to the one-dimensional model in Section 4.2. To be complete, potential pedestrians can walk in one direction or the opposite direction only. In contrary to the article [1], the position in the space is under these conditions single component vector $x \in \mathbb{R}$. The variable A denotes the observed area, i.e. interval of space between locations a and b, where $a, b \in \mathbb{R}$. The function $p(x, blur)$ denotes the density distribution in the area A.

As discussed in the article, using kernel distribution theory, definition of density can be expanded to

$$\rho = \frac{N}{|A|} = \frac{\int_A p(x, blur)dx}{|A|} = \frac{\int_A \sum_{\alpha=1}^{N} p_\alpha(x, blur)dx}{|A|} = \frac{1}{N}\sum_{\alpha=1}^{N} \frac{\int_A p_\alpha(x, blur)dx}{|A|}\,, \tag{2.5}$$

where p_α denotes the individual density distribution generated by each pedestrian $\alpha \in \{1, 2, \ldots, N\}$. Parameter $blur \in \mathbb{R}^+$ denotes kernel parameter, which is size of affected area $A_\alpha \subset A$ by pedestrian α, i.e.

$$A_\alpha = \{x \in A \,|\, p_\alpha(x, blur) > 0\}\,. \tag{2.6}$$

Due to the fact that individual density distribution is normalised, i.e.

$$\int p_\alpha(x)dx = 1\,, \tag{2.7}$$

the following relation can be interpreted as the count of pedestrians in the whole are A,

$$\mathrm{R} = \int_A p(x, blur)dx = \int_A \frac{1}{N}\sum_{\alpha=1}^{N} p_\alpha(x, blur)dx\,. \tag{2.8}$$

In Section 4.2, the previous relation is considered to be the mass of agents, extending the count with the option of a smaller area resulting in the integration of parts of the individual density distribution of a pedestrian.

There is a variety of kernel types used as individual density distribution in Equation (2.5). The standard approach is the **Dirac delta function** that represents a point approximation of the pedestrian distribution, denoted as

$$p_\alpha(x) = \delta_{x,x_\alpha} \,. \tag{2.9}$$

Another popular kernel example is the **Gaussian kernel**, applied in the following example,

$$p_\alpha(x, blur) = \frac{1}{2\pi blur^2} \exp\left(-\frac{(x - x_\alpha)^2}{2blur^2}\right) \,. \tag{2.10}$$

An additional possible kernel example, used in Section 4.2, is a asymmetric triangular kernel function, where $x_\beta \in \mathbb{R}$ denotes the position of pedestrian β, and constant $C = \frac{2}{c_{\text{back}} + c_{\text{front}}}$ denotes a normalization coefficient, i.e.

$$\mathrm{K}(x; c_{\text{back}}, c_{\text{front}}) = \begin{cases} C \cdot (1 - \frac{|x_\beta - x|}{c_{\text{back}}}) & \text{for } z \in (x_\beta - c_{\text{back}}, x_\beta)\,, \\ C \cdot (1 - \frac{|x_\beta - x|}{c_{\text{front}}}) & \text{for } z \in (x_\beta, c_{\text{front}})\,, \\ 0 & \text{for } z \notin (x_\beta - c_{\text{back}}, x_\beta + c_{\text{front}})\,. \end{cases} \tag{2.11}$$

where asymmetry is achieved by replacing parameter *blur* with two different parameters.

This kernel is selected due to the interpretative and computational simplicity. The capabilities of such a kernel should be sufficient, because the article [1] provides a study proving generality of the concept, i.e. after proper parameterization, kernel methods incorporate the non-kernel approaches in the results. It is noteworthy that the effect of kernel parameterization on the results was not fully explored, therefore there exists no general approach to solve the parameterization.

Chapter 3

Pedestrian Flow in Recreational Scenarios

This chapter provides an overview of pedestrian behavior phenomena, observations, and measured data. Additionally, the chapter discusses the usability of the observations for the calibration of model parameters that is presented in Section 5.1.

Initially, selected phenomena and mostly data of non-recreational origin are presented. This is because only fragmentary pieces of evidence exist for unstressful flow, and a research greatly focuses on emergent scenarios and urban traffic. Although these environments differ from recreational scenarios, they share the fundamentals of pedestrian movement. Further, evaluation methods for walkway capacity and metrics for measuring the quality of the pedestrian experience are introduced. Pedestrian Level of Service (PLOS) as an evaluation of pedestrian flow is presented, and further extensions are discussed. In addition, the following text introduces works related to pedestrian movement in similar scenarios. Articles that estimate trail capacity in national parks are briefly presented and agent-based models that simulate recreational pedestrian movement are compared with the model introduced in this thesis. Finally, recent approaches used to simulate pedestrian movement are briefly debated.

3.1 Empirical Findings

Due to the fact that each pedestrian has different movement characteristics, published results from different authors vary. The following text introduces selected results presented in the paper [18] that compares data from different authors and introduces common phenomena and pedestrian behaviour. Despite the submission date, the paper is still cited abundantly today and the relevance of the presented results does not decline due to integrity of pedestrian behavior. The article states that the most representative data for comfort situations, such as recreational walks in the national park, come from the paper [19] which takes measurements on student campuses. It is noteworthy, that pedestrian behavior may differ significantly depending on the country, culture and social group.

The **distribution of unimpended pedestrian speed** is commonly in the range of $0.75 - 2$ ms^{-1} with standard deviation $\sigma = 0.26$ ms^{-1}. According to the report [20], the average speed in leisure situations is approximately 1.2 ms^{-1}. In contrast to vehicle traffic, pedestrian speed is unlimited and pedestrians can accelerate and slow down instantly.

The report [20] further states that the size of a pedestrian body is on average 50×60 cm for standing areas and 0.75×0.75 m for walking areas. Further, the article [18] states that the distance to see the person from the head to the toe is 2.1 m and the lateral width between

strangers is 75 cm and within members of a group 65 cm.

The observations presented work with the quantity **space allocation per pedestrian**, therefore the relationship between the allocation of space, velocity v, which is defined in Definition 1.11, and flow J, defined in Definition 2.2, is introduced in the following equation,

$$\text{space allocation} = \frac{v}{J} \left[m^2/\text{ped}\right] , \tag{3.1}$$

where symbol ped represents 1 pedestrian.

In the context of Section 4.2, following formula describes the relationship between space allocation per pedestrian and the agent mass in the perceived surroundings, where area denotes the area of the perceived surroundings.

$$\text{agent mass} = \frac{\text{area}}{\text{space allocation}} \tag{3.2}$$

More space is required for faster movements. Slow walkers lose the ability to maintain their chosen speed in space allocation $1.4 - 1.7$ m^2/ped. On the contrary, fast walkers lose the ability to maintain their chosen speed in space allocation $2.8 - 3.7$ m^2/ped.

Jam is a flow situation, in which speed v_α of at least one pedestrian α reaches $v_\alpha = 0$. Under jammed conditions, there is approximately 60 cm lateral spacing. Lifetime of a jam distribution is an important quantity in capacity estimation. Most frequent jam reason is a presence of a **bottleneck**, which is narrowed section of a trail, such as in Section 4.3.

Headway distribution is next important quantity that characterizes pedestrian flow. It is a distribution of spacing between pedestrians with the same direction. Details can be found in the paper [21] that deals with headway distribution for the model used in Section 4.3.

The article [18] further mentions that evasion from fixed objects starts at 5 m and distance from moving object to make evasive actions ranges from $0.6 - 5.2$ m based on the speed of the agent and space allocation conditions. In space allocation 3.7 m^2/ped the distance in which evasive actions are taken in the face of imminent collision is 0.6 m while in space allocation 30 m^2/ped the distance is 2.1 m.

Passing slower pedestrians is unrestricted in space allocation above 3.3 m^2/ped and is impossible without confrontation between pedestrians below space allocation of 1.7 m^2/ped. Chance of conflicts between pedestrians drops to zero around space allocation above 4.2 m^2/ped. Mostly, the chance of conflicts under this space allocation is around $50 - 60\%$. It is noteworthy that the model presented in Chapter 4 is collision-free.

The following observations of the average speed ratio of pedestrians in relation to space allocation serve as a source for the appropriate calibration of the fundamental diagram parameters presented in Section 4.2. The observations are supplemented with Table 3.1.

The paper [18] merges results from different author. It concludes that in space allocation 9.3 m^2/ped the average speed is about $96 - 97\%$ of optimal speed. Further, average speed in space allocation 3.7 m^2/ped is observed to be $90 - 93\%$ of optimal speed, in space allocation 1 m^2/ped to be $64 - 75\%$ of optimal speed, and at $0.4 - 0.9$ m^2/ped to be $27 - 50\%$ of optimal speed. To be complete, conditions bellow space allocation 1 m^2/ped lead inevitably to stoppages.

Lane formation is a phenomenon that often solves the jam problem at the bottleneck. Pedestrians create one long line on both sides of the bottleneck. Each line has one leader that determines the speed of a line and all pedestrians in the line move fluently through the bottleneck. Similar phenomena to lane formation is **platooning** and bunching, which occurs already under 50 m^2/ped. The important observation about this phenomena is that flow not categorised as constrained may belong to the constraint conditions when platooning occurs.

It is noteworthy that pedestrians tend to accelerate or slow to avoid walking side by side strangers. This leads to the **checkboard spacing** pattern phenomenom, which occurs around space allocation 5.6 m^2/ped.

Table 3.1 Pedestrian behavior related to available space, taken from [18]

Space Allocation [m^2]	Flow	Average Speed	Choice of Speed	Crossing or Reverse Movement	Conflicts	Passing
$0.2 - 0.5$	Erratic, on the verge of complete stoppage	Shuffling only	None	Impossible	Physical contact Unavoidable	Impossible
$0.5 - 0.7$	Attains a maximum in traffic streams under pressure	Mostly shuffling	None, movement only with crowd	Most difficult	Physical contact probable, conflicts unavoidable	Impossible
$0.7 - 1$	Attains a maximum in more relaxed traffic streams	67% of free flow	Practically none	Severely restricted with collisions	Physical contact probable, conflicts unavoidable	Impossible
$1 - 1.4$	$65 - 80\%$ of maximum capacity	75% of free flow	Restricted, constant adjustments to gait necessary	Severely restricted with conflicts	Unavoidable	Rarely possible without touching
$1.4 - 1.7$	$56 - 70\%$ of maximum capacity	80% of free flow	Restricted except for slow walkers	Restricted with conflicts	Highly probable	Rarely possible without touching
$1.7 - 2.3$	$\leq 50\%$ of maximum capacity	$\geq 80\%$ of free flow	Partially restricted	Possible with conflicts	Highly probable	Difficult without abrupt maneuvers
$2.3 - 3.7$	$\leq 33\%$ of maximum capacity	Approaching free flow	Occasionally restricted	Possible with occasional conflicts	Probably 50% of the time	Possible with interference
≥ 3.7	$\leq 20\%$ of maximum capacity	Virtually as chosen	Virtually unrestricted	Free	Maneuvering needed to avoid conflicts	Free with some maneuvering

3.2 Evaluation of Capacity and Experience

The primary motivation for simulating pedestrian movement in the Luzen Valley is the aim to estimate capacity, which is fundamentally determined by a pedestrian experience. The research presented in this section aims to identify pedestrian flow characteristics, capacity estimate methods, and methods to measure the pedestrian experience to achieve a sufficiently sparse pedestrian flow that does not induce stress for pedestrians.

The greatest motivation for the simulation of a pedestrian movement in the Luzen Valley is the estimate of capacity, which is fundamentally determined by the pedestrian experience. The research of pedestrian flow characteristics, capacity estimation and measuring experience aims to achieve sufficiently sparse pedestrian flow to induce no stress for the pedestrians.

As previously stated, the quantitative characteristics of pedestrian flow include the space allocation and the average speed and flow rate. These characteristics are used to evaluate the capacity of pedestrian facilities. Additionally, the report [20] presents qualitative measures that determine the experience of the pedestrians visiting the recreational facility, such as

- freedom in speed choice,
- ability to pass slower pedestrians,
- ability to cross pedestrians,
- ability to walk in the reverse stream,
- maneuver without conflicts,
- freedom to stop and enjoy the view without impeding others or be impeded.

In addition to the previous section, observations in the article [18] of space allocation directly related to the capacity are presented. The allocation of space under 0.66 m^2/ped is definitely unacceptable. Capacity of urban facilities, such as shopping centres and community centres, are designed with the space allocation around 1 m^2/ped. From the pedestrian view, the space allocation $1.7 - 3.3$ m^2/ped is tolerable in heavily used facilities. Although around 3.5 m^2/ped pedestrians have a free choice of speed and direction, they continuously interact with each other and are aware of others, which decreased the quality of the experience. Furthermore, the recommended space allocation above 12 m^2 is recommended for the narrow sidewalks, such as the section of the model presented in Section 4.3.

As the report [22] discusses, in bidirectional flow, such as in the model introduced in this thesis, the capacity strongly depends on the flow distribution of the two directions. Each stream occupies a share of a walkway proportional to its share in total flow. Hence, the directional imbalance increases the reduction in capacity.

It is noteworthy that the article [18] divides the pedestrian flow into seven categories. The categories are displayed in Table 3.2. As mentioned in the previous section, pedestrian flow with platoons may be constrained in different conditions. In addition to average flow, table displays characteristics for the pedestrian flow with platoons.

Level of Service (LOS) is a quantitative measure of traffic conditions that divides traffic flow into several groups, typically A-F, based on density. Unlike in highway traffic design, operating at maximum capacity is not desirable in pedestrian facilities. **Pedestrian Levels of Service** (PLOS) were introduced to define acceptable pedestrian behaviour under different conditions. To be complete, selection of particular level is a matter of judgment and policy.

The article [20], which focuses on capacity estimate for pedestrian walkways in Parks Victoria, USA, largely revolves around PLOS discussion and capacity parameters such as pedestrian space allocation and walking speed. In addition, the paper presents Table 3.3 that displays an example of LOS categories based on space allocation and interperson spacing. Similarly, the paper [23]

Table 3.2 Characteristics of average flow and flow in platoons, taken from [18]

	Average Flow		**Possible Flow in Platoons**	
Quality of Flow	Space per Pedestrian [m^2]	Flow Rate [ped/min/m]	Space per Pedestrian [m^2]	Flow Rate [ped/min/m]
Open	More than 49.238	Less than 0.5	More than 49.238	Less than 0.5
Unimpeded	49.238 to 12.077	0.5 to 2		
Impeded	12.077 to 4.645	2 to 6	5.574 to 3.716	4.645 to 6
Constrained	3.716 to 2.230	6 to 10	3.716 to 2.230	6 to 10
Crowded	2.230 to 1.486	10 to 14	2.230 to 1.486	10 to 14
Congested	1.486 to 1.022	14 to 18	1.486 to 1.022	14 to 18
Jammed	0.186 to 1.022	18 to 25	Less than 1.022	More than 18

focuses on modeling PLOS for various pedestrian walking facilities in India through pedestrian behavioral responses. Further, the report [22] describes the methods used to calculate PLOS and presents several PLOS models.

Table 3.3 Level of Service criteria for pedestrian traffic, taken from [20]

LOS	**Space [m^2/ped]**	**Flow Rate [ped/min/m]**	**Average Speed [m/s]**
A	> 5.6	< 14	> 1.3
B	$3.7 - 5.6$	$14 - 21$	$1.27 - 1.3$
C	$2.2 - 3.7$	$21 - 33$	$1.22 - 1.27$
D	$1.4 - 2.2$	$33 - 49$	$1.14 - 1.22$
E	$0.75 - 1.4$	$49 - 60$	$0.75 - 1.14$
F	< 0.75	-	< 0.75

The values and recommendations presented in this section can be valuable for the calibration of the model parameters conducted in Section 5.1, and evaluation of the experiments discussed in Chapter 6.

From the six levels shown in Table 3.3, the paper [20] recommends LOS level D as a capacity for day use facilities in national parks. This corresponds to space allocation $1.4 - 2.2$ m^2/ped, flow rate $33 - 49$ ped/min/m and average speed around $1.14 - 1.22$ m/s. Furthermore, Table 3.4 displays recommended walkway widths for LOS level D, which can be used for calibration of parameters in model introduced in Section 4.3.

Table 3.4 Recommended walkway widths for LOS D given different flow rates, taken from [20]

Flow Rate [ped/min]	**Speed [m/min]**	**Space [m^2]**	**Recommended Width [m]**
10	$68.4 - 73.2$	$0.15 - 0.14$	1.5
25	$68.4 - 73.2$	$0.37 - 0.34$	1.5
50	$68.4 - 73.2$	$0.73 - 0.68$	1.5
100	$68.4 - 73.2$	$1.46 - 1.37$	$2.44 - 2.28$
150	$68.4 - 73.2$	$2.19 - 2.05$	$3.65 - 3.42$
200	$68.4 - 73.2$	$2.92 - 2.73$	$4.87 - 4.55$

The article [18] recommends that expected visitor flows can be estimated using data from existing facilities, which have similar distances to the parking lots, population centres and share similar physical characteristics. This estimate can be used in the subsequent studies and is not incorporated in this thesis. In addition, the paper [20] recommends the arrival rate of pedestrians to be less than 50 ped/min for narrow walkways, such as the bottleneck in Luzen Valley.

It is evident that PLOS is not without its shortcomings. For instance, an increase in traffic volume is associated with an improved PLOS score. Additionally, PLOS demonstrates insensitivity to various environmental factors along walkways, such as pavement type, surrounding

natural elements, and the presence of parking lots. Other concepts that aim to quantify the performance of pedestrian facilities were published and compensate for the imperfections. It is crucial to acknowledge that these concepts are primarily applicable to urban settings and are not particularly pertinent to the analysis of pedestrian flow in the Luzen Valley.

Quality of Service (QOS) evaluates the pedestrian flow from the pedestrian's point of view. In addition to PLOS, this concept analyses safety, intersection types, and availability of urban facilities. Table 3.5 displays example of LOS levels for pedestrian delay at crossings. In recent years, municipal offices implement **Multimodal level of service** (MMLOS) to evaluate highways, roads, sidewalks, and intersections in cities. This concept includes evaluation of pedestrian and drivers experience and capacity.

The previously presented observations related to average pedestrian speed are summarized in Figure 3.1. The parameter calibration introduced in Section 5.1 is based on this summary.

Table 3.5 LOS levels for pedestrian delay at crossings, taken from [22]

LOS	Delay [s/p]	Comment
A	$0-5$	Usually no conflicts
B	$5-10$	Occasionally delay due to conficts
C	$10-20$	Delay noticeable to pedestrians, but not inconveniencing
D	$20-30$	Delay noticable and irritating, likelihood of risk taking increased
E	$30-45$	Delay approaches tolerance level, risk-taking behavior likely
F	>45	Delay exceeds tolerance level, high likelihood of pedestrian risk taking

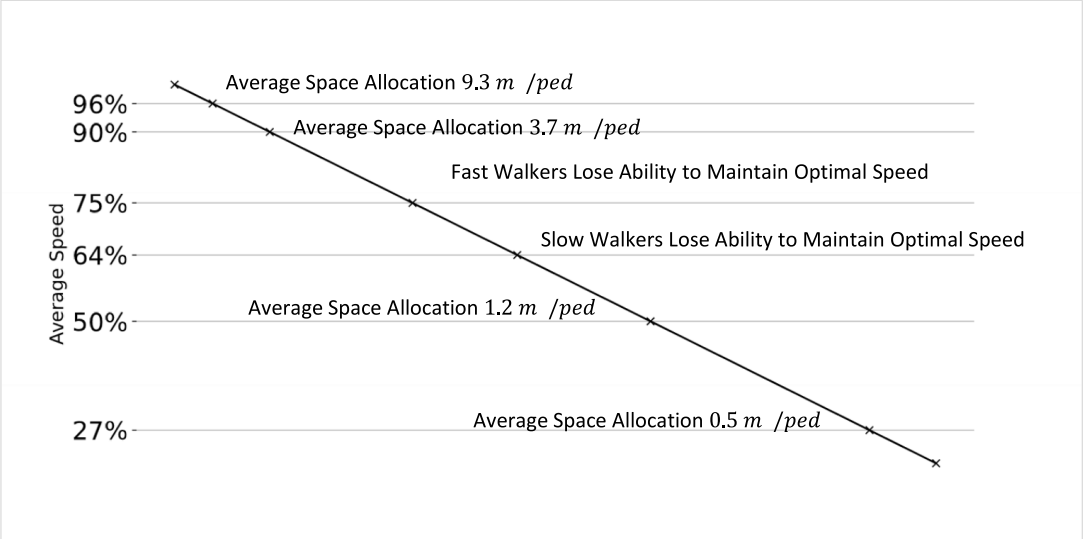

Figure 3.1 Empirical findings related to average pedestrian speed

3.3 Related Work and State of the Art

The subsequent work will involve experiments with restriction of pedestrians movement along the trail. For this purpose, the book [24], which synthesizes social science literature to explain principles of outdoor recreation management, may be of use. Similarly, the article [25], which presents case studies of pedestrian movement simulation within national parks and discusses its

contributions to sustainable tourism, may serve as a source of inspiration for subsequent phases of this project.

Further, the article [26] details the historical development of simulation models for recreational pedestrian movement. It highlights the role of simulation in understanding current visitor patterns and estimating the impact of pedestrian flows on management objectives. Additionally, the paper identifies challenges in computer simulation within outdoor recreational management and offers recommendations for modeling pedestrian movement. This paper can be valuable source for the improvement of the current model in the subsequent work.

Similarly, the article [27] provides a historical overview of pedestrian simulation in recreational settings, which offers insights into future directions. Meanwhile, the paper [28] introduces benchmark tools for agent-based models and evaluates common schemes in agent-based models for pedestrian movement, uncovering inherent imperfections. These findings can be used in the subsequent work to evaluate the agent-based model of the trail.

In addition, the doctoral thesis [29] provides a comprehensive methodology for modeling pedestrian behavior, relevant for shaping desired pedestrian behavior within the model in the subsequent work.

The article [30] investigates the influence of restriction conditions on pedestrian flow capacity within facilities. The findings hold relevance for estimating capacity along the bottleneck section in the subsequent work. Additionally, the article underscores limitations in utilizing a singular fundamental diagram to describe pedestrian flow across diverse path types. Various studies such as [31, 32, 33] analyze real-world urban traffic observations. Their findings may offer additional perspectives on capacity dynamics.

Further, the paper [34] presents a laboratory experiments that were conducted to examine bidirectional pedestrian walkway flows and its impact on the fundamental diagram. Their results reveal a significant differences in flow rates between unidirectional and bidirectional streams, which can be compared with the current model in the subsequent work.

Related articles [35, 36], which similarly focus on managing visitors in protected areas, present a simulations of pedestrian movement in national parks using software NetLogo[1]. Despite their focus on visitor experiences, model created in this software may be compared in the subsequent work with the mathematical model introduced in this thesis. Similarly, the agent-based models presented in the articles [37, 38, 39, 40, 41] may be compared with the model in the subsequent work. They share various characteristics with the model introduced in this thesis.

The article [37] develops an agent-based model that simulates tourist flow and decision-making scenarios based on various constraints. Further, the article [38] proposes a rule-driven agent-based model using geographic information systems to analyze pedestrian interaction with the environment. The paper similarly aims to enhance recreational management in national parks. Meanwhile, the paper [39] introduces a collision-free speed-based model for pedestrian flow, and examines influence of walking preferences on pedestrian movement. The article [40] combines microscopic and macroscopic rules to simulate complex pedestrian behavior. Finally, the article [41] employs a cellular automaton model to simulate bidirectional movement, and compares the results with empirical data.

In recent years, there has been a lot of studies employing a data-driven approach to model pedestrian behavior. For instance, the article [42] introduces model simulating pedestrian trajectories using recurrent neural networks. It states that the data-driven approach outperforms the knowledge-based models in accuracy, flexibility and efficiency. The authors expect wider application of the data-driven models. However, the neural networks are black-box method with bad interpretability and heavy reliance on the training data. The paper states that neural network interpretability studies might provide a solution.

[1] https://ccl.northwestern.edu/netlogo/

Chapter 4

Model Description

The Luzen Valley Trail commences at Březník and leads to the following pavement in Germany. The trail can be accessed from both Czech and the German side. Initially, the route comprises a path that borders the Luzen Creek. The path is followed by a 150 cm wide timber walkway traversing the bog, which is considered as the bottleneck section of the model. Beyond the timber walkway, another pathway leads to Luzen Mountain in Germany. In the following model, the pathways are consider as the transport sections. Pedestrians can arrive from both ends of the trail. The individual sections are marked on map in Figure 4.1.

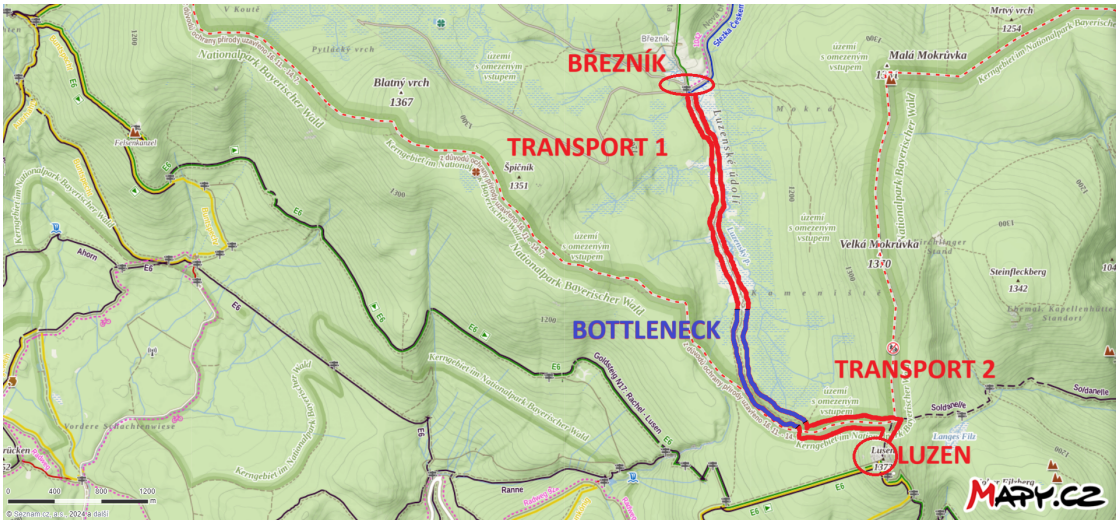

Figure 4.1 Marked sections of the trail on the map, map taken from `mapy.cz`

The trail model comprises three distinct mathematical models, each presented in one section of this chapter. Initially, the chapter introduces random arrivals of agents to the trail. Further, transport section and the bottleneck section are presented. It is important to note that all sections of the model are interconnected.

The current model assumes there are two types of visitors, the visitors that arrive in Březník and are headed to Luzen, and the visitors that arrive in Luzen and are headed to Březník. At both ends of the trail, the journey of agents begins in the transport section that leads to the bottleneck section. After passing the bottleneck section, the agents get through the second transport section to the end of the model and disappear. Figure 4.2 displays an illustration of the model.

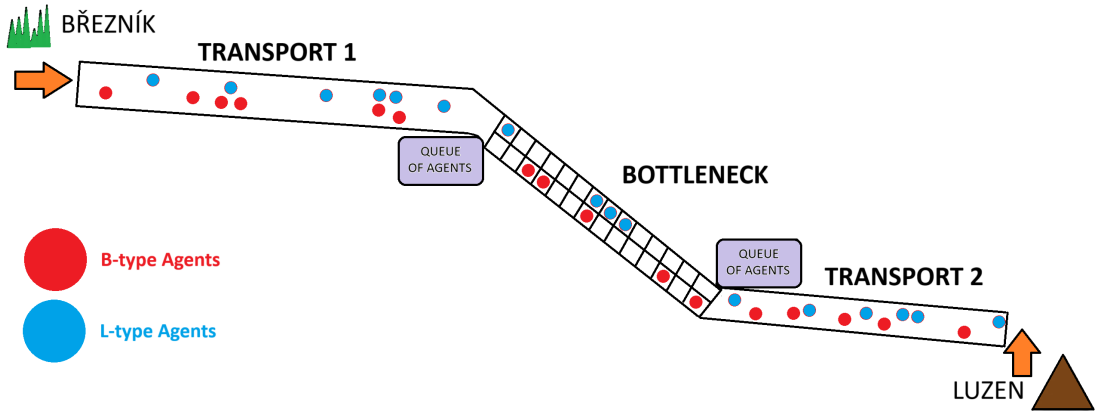

Figure 4.2 Model illustration

4.1 Arrivals

In the following model, the agents are represented by indices. As previously mentioned, there are two types of agents. The agents that begin at Březník belong to the index set B and the agents that begin at Luzen belong to the index set L. The direction of the agents is further referred to as B-type or L-type. Together, the sets form index set of all agents A, i.e.

$$\mathrm{A} = \mathrm{B} \cup \mathrm{L} \tag{4.1}$$

In the following model, each agent $\alpha \in \mathrm{A}$ has individual optimal velocity, denoted as v_α. The direction of agents is distinguished based on the sign of the velocity, where applies the following,

$$v_\alpha \geq 0 \text{ for } \alpha \in \mathrm{B}\,,$$
$$v_\alpha \leq 0 \text{ for } \alpha \in \mathrm{L}\,.$$

The positions of the boundaries of the trail are denoted as x_{start} for Březník, and x_{end} for Luzen. Therefore, the length of the trail is equal to $|x_{\mathrm{end}} - x_{\mathrm{start}}|$. The position of each agent $\alpha \in \mathrm{A}$ at the time t is determined by function $x_\alpha(t)$.

The arrival of each agent $\alpha \in \mathrm{A}$, denoted as t^α_{arr}, is defined as the time when the agent reaches the beginning of the trail, i.e.

$$x_\alpha(t^\alpha_{\mathrm{arr}}) = \begin{cases} x_{\mathrm{start}} & \text{for } \alpha \in \mathrm{B} \\ x_{\mathrm{end}} & \text{for } \alpha \in \mathrm{L} \end{cases} \tag{4.2}$$

The trajectory of agent is considered from the time it reaches the beginning of the trail through the time it reaches the end of the trail, denoted as t^α_{dep}, i.e.

$$x_\alpha(t) \text{ for } t \in \left[t^\alpha_{\mathrm{arr}}, t^\alpha_{\mathrm{dep}}\right] .$$

It is important to note that outside the trajectory, each agent is invisible to the other agents in all sections of the model. To be complete, for each agent's trajectory applies $x_\alpha(t) \in [x_{\mathrm{start}}, x_{\mathrm{end}}]$.

The time period, in which arrivals are possible, begins at 0. The parameter $t^{\mathrm{B}}_{\mathrm{max}}$ limits the time period for arrivals of B-type agents, and the parameter $t^{\mathrm{L}}_{\mathrm{max}}$ limits the time period for arrivals for L-type agents, i.e. applies,

$$t^\alpha_{\mathrm{arr}} \leq t^{\mathrm{B}}_{\mathrm{max}} \text{ for } \alpha \in \mathrm{B}\,,$$
$$t^\alpha_{\mathrm{arr}} \leq t^{\mathrm{L}}_{\mathrm{max}} \text{ for } \alpha \in \mathrm{L}\,.$$

Random arrivals on both ends are modeled by means of a Poisson process, which is subject of Section 1.3. There are both, homogeneous and non-homogeneous, variants possible. Parameters Λ_{B} and Λ_{L} represent the homogeneous arrival rates to the trail for the B-types agents, L-types agents respectively. The non-homogeneous arrival intensity is represented by parameters p_{B} and p_{L}, which denote the probability function at each time, discussed in Equation (1.2), for the B-types agents, L-types agents respectively.

At the beginning of the simulation, times of events are generated for both boundaries of the trail independently. The times of events represent sequence of arrivals to the trail, denoted as T_{B} for the arrivals to Březník, and T_{L} for the arrivals to Luzen.

The cardinality of the set B and L are determined by the size of the corresponding sequences of arrivals. Each agent $\alpha \in \mathrm{B}$ is then assigned the arrival time $t^{\alpha}_{\mathrm{arr}} \in T_{\mathrm{B}}$, and each agent $\beta \in \mathrm{L}$ is then assigned the arrival time $t^{\beta}_{\mathrm{arr}} \in T_{\mathrm{L}}$. It is important to note that Equation (4.2) is the initial condition for the differential equation (4.8) introduced in the following section.

4.2 Transport section

The definition of the model introduced in this chapter contains two identical transport section, one commencing at Březník, one commencing at Luzen. The boundaries of the transport section are denoted $x_{\mathrm{start},\,1}$, $x_{\mathrm{end},\,1}$ for the Březník transport section, $x_{\mathrm{start},\,2}$, $x_{\mathrm{end},\,2}$ for the Luzen transport section respectively. The boundaries of the trail x_{start} and x_{end} are identified with the section boundaries $x_{\mathrm{start},\,1}$, $x_{\mathrm{end},\,2}$ respectively. Illustration of the boundaries is shown in Figure 4.3. To be complete, the B-type agents are moving towards section boundary $x_{\mathrm{end},\,1}$, $x_{\mathrm{end},\,1}$ respectively, and the L-type agents are moving towards $x_{\mathrm{start},\,2}$, $x_{\mathrm{start},\,1}$ respectively.

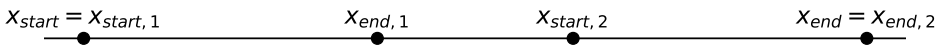

Figure 4.3 Illustration of transport section boundaries

The pavement is expected to be wide enough, approximately 3 m, that microscopic modeling is unnecessary and more efficient and interesting approach is to model the interactions between agents through perception of surrounding agents. The pedestrian flow is modeled in one-dimensional space.

4.2.1 Pedestrian Mass in the Perceived Surroundings

The idea is that each agent generates individual mass around its current position. Each agent reacts to the agent mass in the perceived surroundings by potential decrement of speed. In comparison to density, which is discussed in Section 2.1, the agent mass in the perceived surroundings agents reflects the position of agents on the boundaries of the perceived surroundings. While an agent is located in the perceived surrounding, the mass outside the surroundings is not taken into account, as illustrated in Figure 4.4. If assumed that the mass of pedestrians is identified with the number of pedestrians, the following relation describes relation between mass R and density ρ,

$$\rho(\mathrm{R}) = \frac{\mathrm{R}}{\mathrm{width} \cdot |d_2 - d_1|}$$

where width represents pathway width and $|d_2 - d_1|$ length of the perceived surroundings.

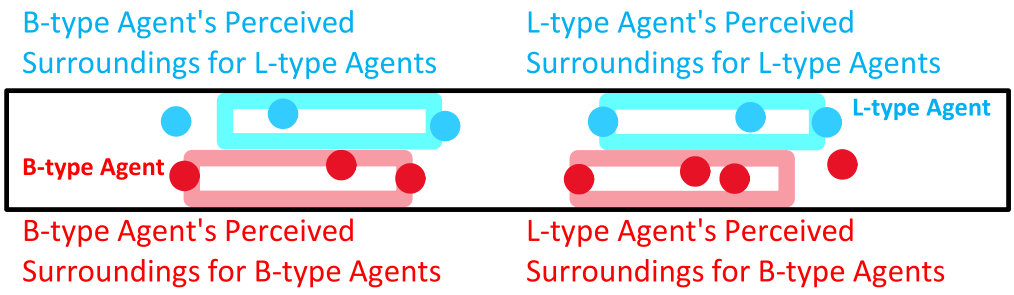

Figure 4.4 Illustration of the perceived surroundings of agents

Despite the values selected in Section 5.1, the model accommodates variability in the perceived surroundings of agents, both walking in the same direction and in the opposite. The motivation is that pedestrians may perceive variously oriented flows differently. The surroundings are determined by the parameters d_1^{dir} and d_2^{dir}, which dictate the range of perception for agents moving in the same direction, and d_1^{opp} and d_2^{opp}, governing the range of perception for agents moving in the opposite direction. It is important to note that the area of the individual mass is expected to be considerably smaller than the perceived surroundings. This fact is especially important for the implementation of the model, discussed in Section 5.2.

At each timestep t, sum of individual masses from the same flow creates a mass profile for each direction, which is illustrated in Figure 4.5. The agent mass in the perceived surroundings is a integral of the mass profile on an interval representing the perceived surroundings of agent. It is important to note that the mass of perceiving agent is not included.

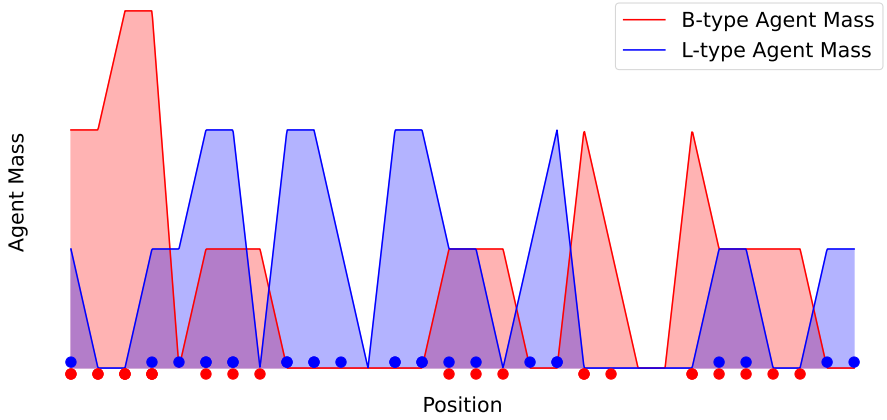

Figure 4.5 Illustration of agent mass profile for both directions

In the following text, the mass generated by agent $\alpha \in \mathrm{A}$ on position y at the time t is denoted as $r_\alpha(y,t)$. As illustrated in Figure 4.4, there are four combinations of types of interacting agents, namely

1. B-type agent perceives B-type agents,
2. B-type agent perceives L-type agents,

3. L-type agent perceives L-type agents,
4. L-type agent perceives B-type agents.

For each combination of types, a different function that expresses the agent mass in the perceived surroundings is introduced in the following formulas (4.3) – (4.6). Each agent contributes to the final sum with the mass that encroaches into the perceived surroundings of the perceiving agent, in other words with non-zero values in the perceived interval of function r_β.

$$R^{\mathrm{B,B}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t; d_1^{\mathrm{dir}}, d_2^{\mathrm{dir}}, d_1^{\mathrm{opp}}, d_2^{\mathrm{opp}}) = \sum_{\beta \in \mathrm{B}\setminus\{\alpha\}} \int_{x_\alpha + d_1^{\mathrm{dir}}}^{x_\alpha + d_2^{\mathrm{dir}}} r_\beta(y,t)\, dy\,, \tag{4.3}$$

where $R^{\mathrm{B,B}}_{\mathrm{per}}$ represents agent mass of B-type agents in the perceived surroundings of the B-type agent.

$$R^{\mathrm{B,L}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t; d_1^{\mathrm{dir}}, d_2^{\mathrm{dir}}, d_1^{\mathrm{opp}}, d_2^{\mathrm{opp}}) = \sum_{\beta \in \mathrm{L}} \int_{x_\alpha + d_1^{\mathrm{opp}}}^{x_\alpha + d_2^{\mathrm{opp}}} r_\beta(y,t)\, dy\,, \tag{4.4}$$

where $R^{\mathrm{B,L}}_{\mathrm{per}}$ represents agent mass of L-type agents in the perceived surroundings of the B-type agent.

$$R^{\mathrm{L,L}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t; d_1^{\mathrm{dir}}, d_2^{\mathrm{dir}}, d_1^{\mathrm{opp}}, d_2^{\mathrm{opp}}) = \sum_{\beta \in \mathrm{L}\setminus\{\alpha\}} \int_{x_{\alpha\,\mathrm{L}} - d_2^{\mathrm{dir}}}^{x_\alpha - d_1^{\mathrm{dir}}} r_\beta(y,t)\, dy\,, \tag{4.5}$$

where $R^{\mathrm{L,L}}_{\mathrm{per}}$ represents agent mass of L-type agents in the perceived surroundings of the L-type agent.

$$R^{\mathrm{L,B}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t; d_1^{\mathrm{dir}}, d_2^{\mathrm{dir}}, d_1^{\mathrm{opp}}, d_2^{\mathrm{opp}}) = \sum_{\beta \in \mathrm{B}} \int_{x_{\alpha\,\mathrm{L}} - d_2^{\mathrm{opp}}}^{x_\alpha - d_1^{\mathrm{opp}}} r_\beta(y,t)\, dy\,, \tag{4.6}$$

where $R^{\mathrm{L,B}}_{\mathrm{per}}$ represents agent mass of B-type agents in the perceived surroundings of the L-type agent.

To be complete, the total agent mass in the perceived surroundings of agent $\alpha \in \mathrm{A}$ at the time t, denoted as R_α, is a sum of the mass perception from the both flows. The motivation is that at each moment, the surrounding pedestrians are objects with the same impact, i.e.

$$R_\alpha(\{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t) = \begin{cases} R^{\mathrm{B,B}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t) + R^{\mathrm{B,L}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t) & \text{for } \alpha \in \mathrm{B}\,, \\ R^{\mathrm{L,L}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t) + R^{\mathrm{L,B}}_{\mathrm{per}}(\alpha, \{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t) & \text{for } \alpha \in \mathrm{L}\,. \end{cases} \tag{4.7}$$

4.2.2 Motion Description

As mentioned in Section 1.4, the velocities $\dot{x}_\alpha(t)$ of agents at the time t are determined by the following coupled differential equations,

$$\dot{x}_\alpha(t) = \mathrm{V}_\alpha(\{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t) \text{ for } \alpha \in \mathrm{A}\,, \tag{4.8}$$

where the speed-governing function V_α depends on the positions of active agents and the time t. The final solution of the equations are the trajectories $x_\alpha(t)$ for each agent $\alpha \in \mathrm{A}$.

The explicit Euler method, described in Equation (1.14) from Section 1.4, approximates the discrete trajectories

$$x_\alpha(t_{n+1}) \text{ where } t_0 = 0 \wedge t_{n+1} = t_n + h \wedge n \in \mathbb{N} \text{ for } \alpha \in \mathrm{A}\,.$$

The function V_α is inspired by the velocity-density fundamental diagram, shown on Figure 4.6 from Section 2.1. Instead of density, the function accepts agent mass in the perceived surroundings discussed in the previous subsection, i.e.

$$\mathrm{V}_\alpha(\{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t) = \mathrm{FD}_\alpha(R_\alpha(\{x_\beta(t) \,|\, \beta \in \mathrm{A}\}, t)) \text{ for } \alpha \in \mathrm{B}\,. \tag{4.9}$$

Form of the function FD_α is shown in Figure 4.6 and in the following formula (4.10). Inner parameters $\mathrm{R}_{\mathrm{critical}}$, and $\mathrm{R}_{\mathrm{max}}$, carry similar meaning as in the ordinary fundamental diagram, i.e. critical value of agent mass in perceived surroundings, at which agent begins to slow down, at which the agent stops respectively.

$$\mathrm{FD}_\alpha(R; v_\alpha, \mathrm{R}_{\mathrm{max}}, \mathrm{R}_{\mathrm{critical}}) = \begin{cases} v_\alpha & \text{for } R < R_{\mathrm{crit}}\,, \\ v_\alpha \cdot (1 - \frac{R - R_{\mathrm{crit}}}{R_{\mathrm{max}}}) & \text{for } R_{\mathrm{crit}} \leq R < R_{\mathrm{max}}\,, \\ 0 & \text{for } R \geq R_{\mathrm{max}}\,. \end{cases} \tag{4.10}$$

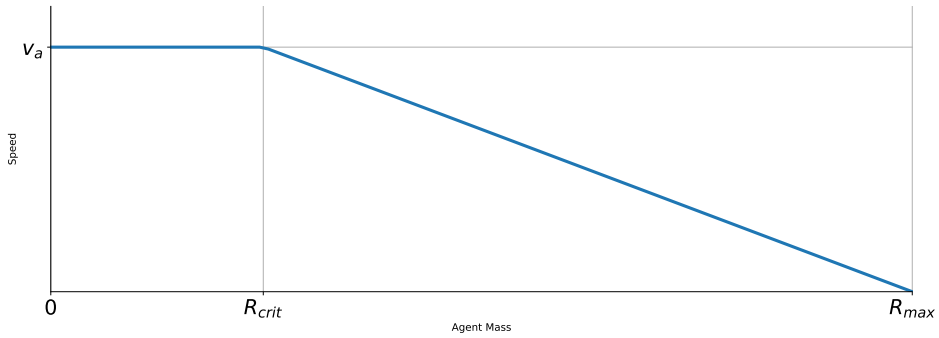

Figure 4.6 Illustration of the velocity-governing function

Based on the discussion from Section 2.2.2, the individual mass generated by agent β, denoted as r_β, is modeled by normalized kernel function with center at $x_\beta(t)$. Normalization ensures that the entire mass of an agent is equal to 1. With asymmetric kernel function there are necessary two different kernel function, one for each walk direction, i.e.

$$r_\beta : (y, t) \mapsto \begin{cases} \mathrm{K}_\mathrm{B}(y - x_\beta(t)) & \text{for } \beta \in \mathrm{B}\,, \\ \mathrm{K}_\mathrm{L}(y - x_\beta(t)) & \text{for } \beta \in \mathrm{L}\,. \end{cases} \tag{4.11}$$

As mentioned in Section 2.2.2, the model uses triangular kernel to represent the individual mass. In walk direction is the outermost point of the kernel given by parameter c_{front} and the back of the agent is given by parameter c_{back}. To be complete, constant C is a scale parameter for normalization, for which it applies

$$C = \frac{2}{c_{\mathrm{back}} + c_{\mathrm{front}}}\,. \tag{4.12}$$

The function K_B, shown on Figure 4.7 on the left, represents mass of agent $\beta \in \mathrm{B}$ at the position z,

$$\mathrm{K}_\mathrm{B}(z; c_{\mathrm{back}}, c_{\mathrm{front}}) = \begin{cases} C \cdot (1 - \frac{|z|}{c_{\mathrm{back}}}) & \text{for } z \in (-c_{\mathrm{back}}, 0)\,, \\ C \cdot (1 - \frac{|z|}{c_{\mathrm{front}}}) & \text{for } z \in (0, c_{\mathrm{front}})\,, \\ 0 & \text{for } z \notin (-c_{\mathrm{back}}, c_{\mathrm{front}})\,, \end{cases} \tag{4.13}$$

and the function $\mathrm{K_L}$, shown on Figure 4.7 on the right, represents mass of agent $\beta \in \mathrm{L}$ at the position z,

$$\mathrm{K_L}(z; c_{\,\mathrm{back}}, c_{\,\mathrm{front}}) = \begin{cases} C \cdot (1 - \frac{|z|}{c_{\,\mathrm{front}}}) & \text{for } z \in (-c_{\,\mathrm{front}}, 0)\,, \\ C \cdot (1 - \frac{|z|}{c_{\,\mathrm{back}}}) & \text{for } z \in (0, c_{\,\mathrm{back}})\,, \\ 0 & \text{for } z \notin (-c_{\,\mathrm{front}}, c_{\,\mathrm{back}})\,. \end{cases} \tag{4.14}$$

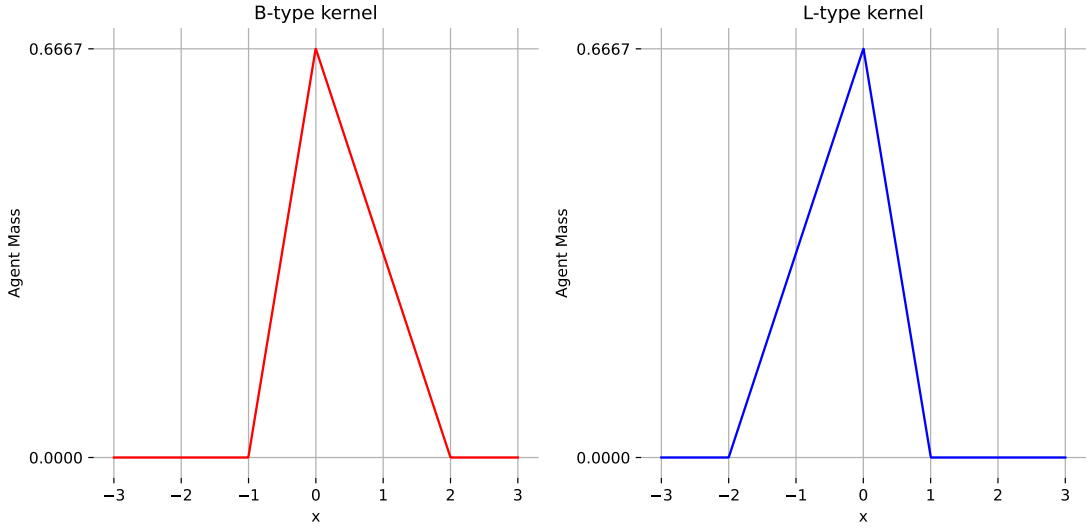

Figure 4.7 Illustration of kernel function, B-type on the left, L-type on the right

4.3 Bottleneck Section

The bottleneck section is the most critical in terms of trail capacity. Due to the fact that this thesis focuses on the transport section of the model, this section is represented by simple cellular automaton called TASEP, which is subject of Section 1.2.1.

Based on the empirical findings presented in Section 3.1, it is estimated that two lines of flow are possible in this section. Each path is represented by a one-dimensional array of cells, where each cell has the capacity for one agent. It is important to note that the two flows do not influence each other.

The scheme of this section is displayed in Figure 4.8. The cell size, denoted as w_{cell}, is identified with the size of pedestrian body, which is one of the model parameters presented further in Table 5.1. The length of this section is the distance between the two transport sections. Number of cells is given by the following formula, where $x_{\mathrm{end},1}$ represents the boundary of the transport section closer to Březník, and $x_{\mathrm{front},2}$ represents the boundary of the transport section closer to Luzen,

$$\text{number of cells} = \left\lceil \frac{|x_{\mathrm{front},2} - x_{\mathrm{end},1}|}{w_{\mathrm{cell}}} \right\rceil . \tag{4.15}$$

In contrast to the previous section, the time is considered to be continuous in the simulation. Continuous time is simulated using the continuous update strategy, described in Section 1.2.1. After each update, each agent $\alpha \in \mathrm{A}$ waits until next update for the time period $t^{\alpha}_{\mathrm{wait}}$, for which it applies

$$t^{\alpha}_{\mathrm{wait}} = X_\alpha \text{ where } X_\alpha \sim \mathrm{Exp}\,(p_\alpha)\,. \tag{4.16}$$

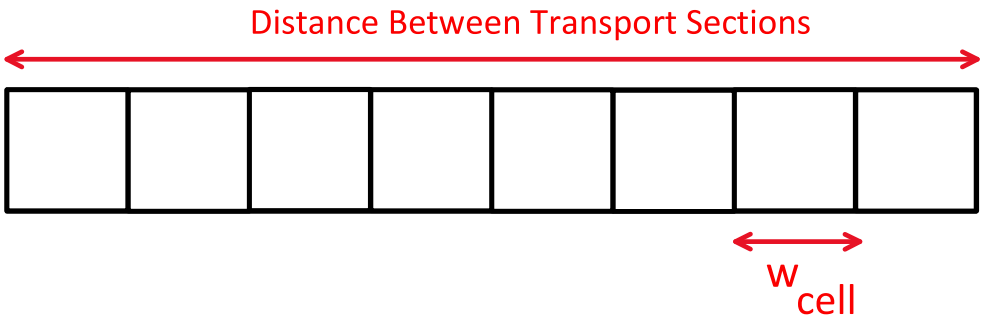

Figure 4.8 Illustration of the bottleneck section

Updates are performed sequentially, in ascending order according to the time of the next update. The transition from discrete to continuous time is carried out by moving agents leaving the preceding section of the model to the queue. The first update time $t^{\alpha}_{\text{update(1)}}$ for each agent $\alpha \in \text{A}$ is given by the following formula, where time t denotes the moment, at which the agent appears at the front of the queue.

$$t^{\alpha}_{\text{update(1)}} = t + t^{\alpha}_{\text{wait}} \,. \tag{4.17}$$

During the update of each agent, if the subsequent cell is not occupied, the agent proceeds to the next cell. Otherwise, the agent waits the subsequent update period. Either way, agent generates new update time. It is expected that each agent $\alpha \in \text{A}$ will take on average $\frac{w_{\text{cell}}}{v_{\alpha}}$ to move to the next cell. Therefore, the parameter of the exponential distribution from Equation (4.16), $p_{\alpha} = EX_{\alpha}^{-1}$, is derived from the following,

$$EX_{\alpha} = \frac{w_{\text{cell}}}{v_{\alpha}} \tag{4.18}$$

$$p_{\alpha} = \frac{v_{\alpha}}{w_{\text{cell}}} \,. \tag{4.19}$$

Chapter 5

Implementation of the Simulation Tool

This thesis project is publicly accessible via personal GitHub repository[1]. The simulation tool is a `C++` program that implements the model introduced in Chapter 4. The program performs both, a simulation, and saving logs for the output analysis. Moreover, the repository comprises `Python` scripts, designed to execute simulations across various parameter sets. In addition, a `Jupyter notebook` crafted to validate the implementation is present in the repository. To add a visual dimension to the exploration, a `Python` script within the repository generates animations depicting the trajectories derived from the simulations.

As displayed in Figure 5.1, a snapshot reveals the trajectories of simulated agents from one such simulation. The x-axis represents the positional data, while each column represents an individual agent, differentiated by color. The red color indicates B-type agents and the blue color indicates L-type agents. Notably, the column height in the transportation section correlates with the current speed of agent. A timer, positioned in the upper left corner, provides temporal context. Additionally, the bottleneck section features a display showing the number of agents queuing at the given time on each side of the bottleneck.

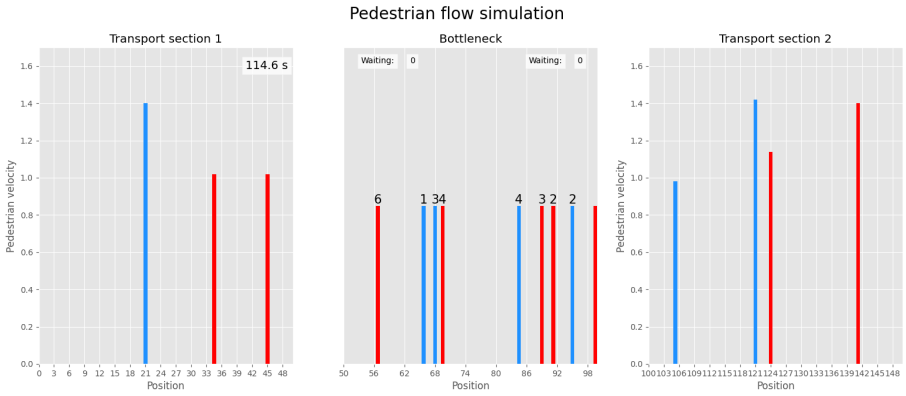

Figure 5.1 Screenshot from the animation of the simulation progress

Initially, Section 5.1 provides an overview of the model parameters. Moreover, the section discusses the calibration of the parameters. In Section 5.2, the groundwork is laid for the `C++`

[1] `https://github.com/tomnovota/bachelor-thesis`

project responsible for realizing the model introduced in this thesis. This section identifies the essential components of the project and explains their roles and interactions. Moreover, functions pivotal to the project functionality are examined in more detail. It is noteworthy that the functionality of the solution is validated in Section 6.1. Finally, Section 5.3 explains how to use the tool. Specifically, the section describes how to pass parameters to the program and run the simulation. Additionally, it defines the format of output files.

5.1 Parameter Overview and Calibration

The model introduced in the previous chapter contains various parameters. Initially, this section briefly presents all parameters and discusses calibration of these parameters. In addition, this section introduces an overview of the simulation parameters in Table 5.1, which also displays the selected values. It is noteworthy that the tool is designed to set all the parameters through a parameter file. The way in which the parameter file is handled is the subject of Section 5.3.

The parameter timestep h represents the discretization of the time, explained in Equation (1.14). The choice of value for this parameter is subject to two criteria. First, the discretization has to be fine enough to ensure smooth behavior of the model. Second, as the value for this parameter decreases, the computational complexity of the whole simulation increases significantly. The value 0.1 s is considered to be fine enough while preserving a reasonable computational efficiency of the simulation.

Further, there are pedestrian parameters, such as the size of the body ellipse and the speed distribution. Based on the analysis in Section 3.1, the body size of pedestrian is considered to be 0.75×0.75 m in this thesis. Moreover, the speed distribution is approximated with trimmed normal distribution with median $\text{speed}_{\text{med}} = 1.2$ m/s, and standard deviation $\text{speed}_{\text{std}} = 0.26$ m/s. The width of the trim on each side is determined by the parameter $\text{speed}_{\text{trim}} = 0.45$ m/s. Figure 5.2 displays the shape of the selected speed distribution.

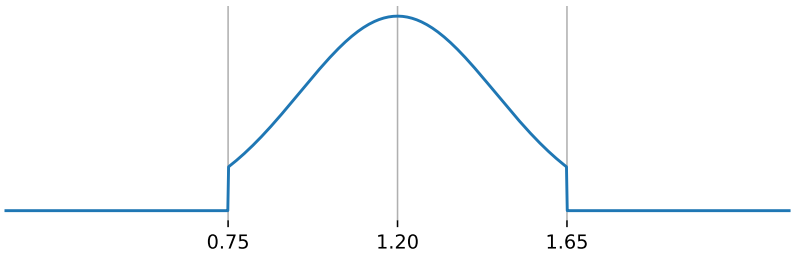

Figure 5.2 Illustration of the selected speed distribution

There are overall 8 parameters that are connected to the mass of agents, introduced in Section 4.2.1. Parameters $d_1^{\text{dir}}, d_2^{\text{dir}}, d_1^{\text{opp}}, d_2^{\text{opp}}$ that represent the perceived surroundings, parameters R_{crit}, R_{max} that are parameters of the fundamental diagram from Equation (4.10), and parameters c_{back}, c_{front} representing the size of the individual mass generated by agents. The following paragraphs discuss the values of the mentioned parameters based on the analysis presented in Chapter 3.

In this thesis, the range of the perceived surroundings is the same for both flows. The perceived range begins at the position of agent, i.e. $d_1 = 0$ m, and ends 5 m in front of the agent, i.e. $d_2 = 5$ m. Furthermore, the space allocation 3.7 m^2/ped is considered to slow down the pedestrians, and conditions with the space allocation 1 m^2/ped are considered to lead to stoppages. The values of the critical and maximum mass are obtained using Equation (3.2). The value *area* in the equation is calculated as the area of a rectangle with dimensions equal to the length of the perceived surroundings and the width of the pavement. As depicted in Figure 5.3,

the area is 15 m^2. Therefore, the parameter values are $R_{crit} = 15/3.7$, and $R_{max} = 15$. Finally, the size of the kernel function is set 0.75 m behind the agent, which is consistent with the size of body ellipse, and $1.2 + 0.75$ m in front of the agent, which is consistent with the body ellipse increased by the average walking speed, i.e. $c_{back} = 0.75$ m, and $c_{front} = 1.95$ m. The idea is that the pedestrians perceive the mass of other pedestrians differently depending on the walking direction.

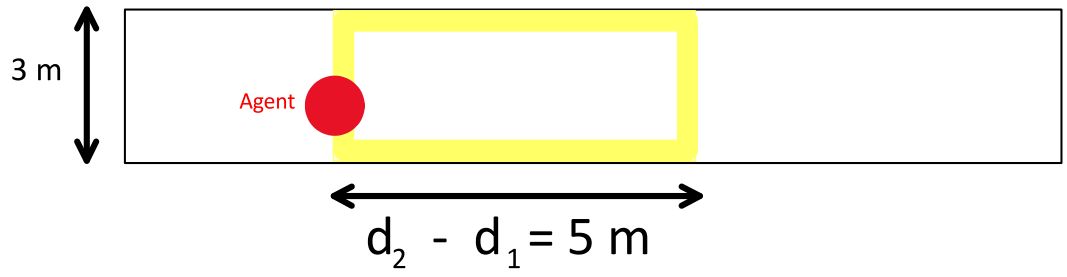

Figure 5.3 Illustration of an area of the perceived surroundings

The only directly selectable parameters are lengths of the sections of the model, i.e. the length of each transport section and the length of the bottleneck section. Based on the distances, the boundaries of both transport sections, denoted as x_{front}, and x_{end}, are calculated. Due to the fact that this thesis does not aim to introduce the final model of the trail, the values of these parameters are artificial. In experiments described in Section 6.2, the length of each section is 150 m. It should be noted that the transport section between Březník and the bottleneck section is called the transport section 1, and the second transport section is called the transport section 2.

The remaining parameters are related to the random arrivals that are subject of Section 4.1. Parameters t^{B}_{max}, and t^{L}_{max} represent the time period of the arrivals, whereas the parameters Λ_B, and Λ_L represent the homogeneous intensity of the arrivals. The aforementioned parameters facilitate the realization of a various scenarios along the trail. Therefore, they are crucial for estimating the capacity of the trail. Based on Table 3.4, the expected flow rate is in order of tens and lower hundreds of pedestrians per minute, which is approximately a range in $0.2 - 30$ ped/s. It is noteworthy that development of the flow with increasing arrival intensity is an essential experiment conducted in Chapter 6. As no interpretable probability of random arrivals has not been introduced, a rational time period of the arrivals is not set yet.

5.2 Implementation of the Agent-based Model

The implementation of the tool adheres to an object-oriented approach. Table 5.2 provides an overview of the classes discussed in this section.

Each pedestrian agent in the simulation is represented by the *CPedestrian* class, which encapsulates properties such as ID, optimal velocity, and current position. Additionally, it stores the arrival time of the agent, the time of passing the initial transport section, and the time of passing the bottleneck section. This class also handles the format for log entries.

The random arrival times are implemented by class *CArrivalGenerator*. The class handles both homogeneous and non-homogeneous arrival intensity. The variable arrival intensity, represented by function $p(t)$ in Equation (1.2), is passed to the class in the form of a polynomial function given by its coefficients. In the absence of the coefficients, the class generates arrivals with homogeneous intensity Λ.

Table 5.1 Overview of Simulation Parameters, grouped by the type of parameter

Parameter Name	Variable	Selected Value	Units
timestep	h	0.1	s
body width	w_{cell}	0.75	m
speed median	$\text{speed}_{\text{med}}$	1.2	m/s
speed deviation	$\text{speed}_{\text{std}}$	0.26	m/s
speed trim	$\text{speed}_{\text{trim}}$	0.45	m/s
critical mass	R_{crit}	15/3.7	mass of pedestrians
maximum mass	R_{max}	15	mass of pedestrians
d1 dir	d_1^{dir}	0	m
d2 dir	d_2^{dir}	5	m
d1 opp	d_1^{opp}	0	m
d2 opp	d_2^{opp}	5	m
c back	c_{back}	0.75	m
c front	c_{front}	1.95	m
distance transport1	$\lvert x_{\text{end, 1}} - x_{\text{start, 1}} \rvert$	150	m
distance bottleneck	$\lvert x_{\text{start, 2}} - x_{\text{end, 1}} \rvert$	150	m
distance transport2	$\lvert x_{\text{end, 2}} - x_{\text{start, 2}} \rvert$	150	m
arrival period B	$t_{\text{max}}^{\text{B}}$	1500	s
arrival period L	$t_{\text{max}}^{\text{L}}$	1500	s
arrival intensity B	Λ_{B}	$0.1 - 30$	ped/s
arrival intensity L	Λ_{L}	$0.1 - 30$	ped/s

The behavior of agents is differentiated based on their location within the trail network. Agents exhibit distinct behaviors in both transport and bottleneck sections. In the transport section, agents maintain a position range for calculation of the agent mass in the perceived surroundings discussed in Section 4.2.1. In the bottleneck section, agents handle the time of the next update.

Each transport section of the model is represented by subclass of the class *CTransport*. Subclass *CTransportB* manages transport from Březník, while subclass *CTransportL* manages transport from Luzen. The class defines the environment parameters such as the positions of the left and right boundaries of the section. It incorporates the fundamental diagram function, given by Equation (4.10), along with critical density and maximum density parameters. Additionally, it stores array of active agents for each walking direction.

The *CMass* class handles calculations of agent mass. The implementation is designed for comparison of different kernels, allowing for the addition of new kernels as subclasses of the *CMass* class. Currently, the implemented kernels include the Diraq and triangular kernels, described in Section 2.2.2, represented respectively by the *CDiraq* and *CTriang* subclasses. No interaction between pedestrians is represented by subclass *CFree*. In addition, the triangular kernel requires storing the parameters $c_{\text{back}}, c_{\text{front}}$, found in Equation (2.11).

The bottleneck section is handled by subclasses of the *CBottleneck* class. Due to the fact that the two walking directions do not cross, two independent instances manage each walking direction within the bottleneck. The subclass *CBottleneckB* serves B-type agents, while subclass *CBottleneckL* serves L-type agents. The lattice is represented by an array of agents. The size of the array is determined by Equation (4.15). Further, class implements a queue of agents waiting to move to the beginning of the bottleneck, and storing an order of agents by time of next update.

Additionally, the *CParameters* class reads parameters from the parameter file, enhancing the configurability of the simulation, while the *CConfig* class oversees simulation control with specialized subclasses for simulation of all sections of the model through *CConfigFull* and transport section-only simulation through *CConfigTransport*.

Implementation of the agent-based model, introduced in Chapter 4, follows a scheme illus-

Table 5.2 Overview of classes present in `C++` project

Class	Description	Subclasses
CArrivalGenerator	generates arrival times	
CPedestrian	represents agent	
CTransport	represents transport section	CTransportB, CTransportL
CMass	represents mass calculations	CDiraq, CTriang, CFree
CBottleneck	represents bottleneck section	CBottleneckB, CBottleneckL
CParameters	reads parameter file	
CConfig	controls simulation	CConfigFull, CConfigTransport

trated in Figure 5.4. Before simulation of the movement, all agents are created, and assigned a arrival time, as described in Section 4.1. Thereafter, the agents are inserted into corresponding reservoir of inactive agents on each boundary of the model, depicted in the scheme.

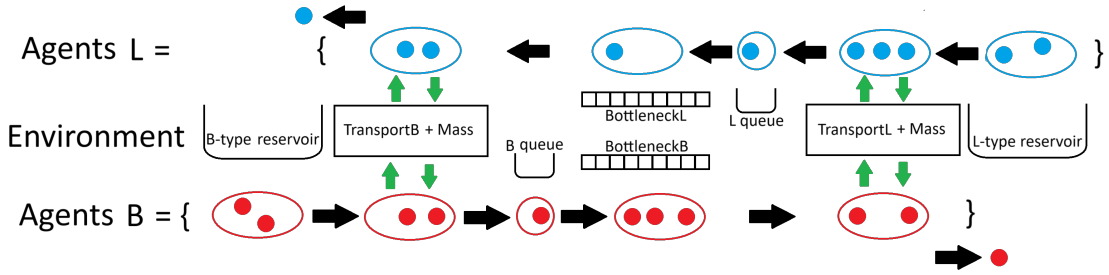

Figure 5.4 Scheme of agent-based model implementation

5.2.1 Crucial Components in Detail

The algorithm for generation of random arrival times, based on Section 1.3, is written in Algorithm 1, where the parameter *period* indicates the time period $t_{\max}$ discussed in Section 4.1, the parameter *lambda* denotes the maximum arrival intensity Λ, and parameter *coeffs* denotes coefficients of polynomial function $p(t)$, representing the non-homogeneous intensity of arrivals discussed in Section 1.3.

Prior to the description of the simulation steps, algorithms representing calculation of the agent mass in the transport section are presented. Figure 5.5 displays a set of possible cases of the overlay of the perceived surroundings and the individual mass. As discussed previously in Section 4.2.1, presented overlays assume that the perceived surroundings is greater than the area of the individual mass. As the figure depicts, the perceived area of this triangular kernel is in each case calculable exclusively by areas of a right-angled triangles.

Algorithm 2, and Algorithm 3, represent simple functions that calculate the area of a right-angled triangle that is part of the back triangle of a triangular kernel, front triangle respectively, given by the distance from the origin. The normalizing coefficient C is explained in Equation (2.11).

Further, follows the implementation of Algorithm 4 and Algorithm 5, which are derived from the previous analysis of various overlay cases. The constants Content_Back, and Content_Front, represent the area of the right-angled triangle where $side = c_{\text{back}}$, $side = c_{\text{front}}$ respectively.

Notably, Algorithm 5 distinguishes itself from Algorithm 4 through mirrored utilization of the concepts of ”back” and ”front”.

The discretization of time in the transport section, which comes from Equation (1.14), results

Algorithm 1 GenerateEvents(period, lambda, coeffs)

procedure GENERATEEVENTS($t_{\max}$, λ, $coeffs$)
    $events\_homo \leftarrow$ Empty array
    while True **do**
        $time \leftarrow X+$ LastEvent($events\_homo$); $X \sim$ Exp (λ)
        if $time > t_{\max}$ **then**
            break
        end if
        AddEvent($events\_homo$, $time$)
    end while
    if not IsEmpty($coeffs$) **then**
        $events\_nonhomo \leftarrow$ Empty array
        for each $time$ in $events\_homo$ **do**
            $prob \leftarrow$ CalculateProbability($time$, $coeffs$) / λ
            if $Y \geq prob; Y \sim Binomial(1, prob)$ **then**
                AddEvent($events\_nonhomo$, $time$)
            end if
        end for
        return $events\_nonhomo$
    else
        return $events\_homo$
    end if
end procedure

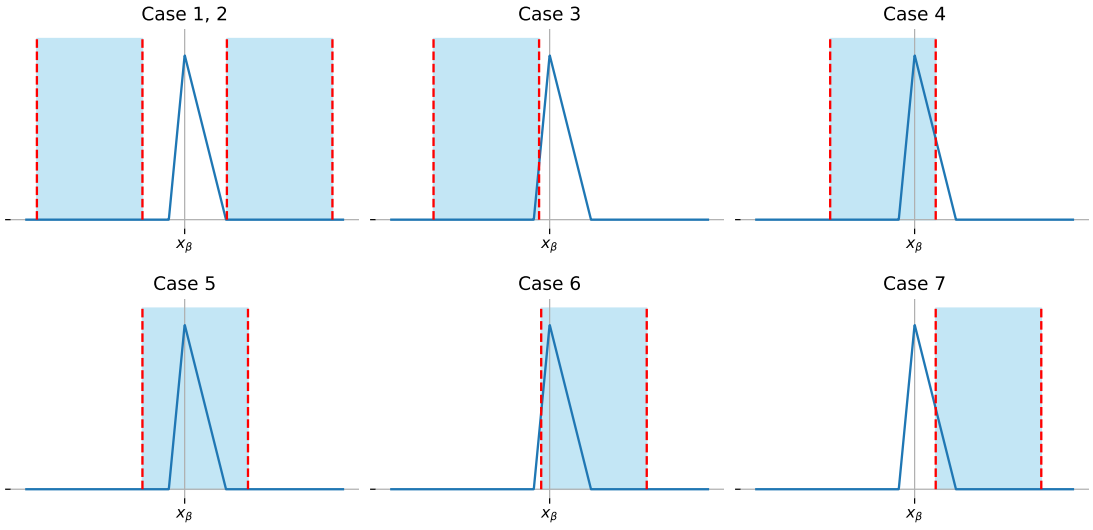

Figure 5.5 Visualization of all possible cases of the overlay of the perceived surroundings and the individual mass represented by triangular kernel

Algorithm 2 CTriangle::TriangContentBack(x)

function TRIANGCONTENTBACK(x)
  $h \leftarrow 2 \cdot (c_{\text{back}} + c_{\text{front}})^{-1}$ // (4.12)
  $height \leftarrow C \cdot (1 - \frac{|x|}{c_{\text{back}}})$
  $side \leftarrow \max(0, c_{\text{back}} - |x|)$
  return $height \cdot side/2$
end function

Algorithm 3 CTriangle::TriangContentFront(x)

function TRIANGCONTENTFRONT(x)
  $h \leftarrow 2 \cdot (c_{\text{back}} + c_{\text{front}})^{-1}$ // (4.12)
  $height \leftarrow C \cdot (1 - \frac{|x|}{c_{\text{front}}})$
  $side \leftarrow \max(0, c_{\text{front}} - |x|)$
  return $height \cdot side/2$
end function

Algorithm 4 CTriangle::GetMassB(x_beta, from, to)

function GETMASSB(x_β, $from$, to)
  $mass \leftarrow 0$
  $i_1 \leftarrow from - x_\beta$
  $i_1 \leftarrow to - x_\beta$
  if $[-c_{\text{back}}, c_{\text{front}}] \cap [i_1, i_2] = \emptyset$ **then**
    return 0 // case 1, 2
  end if
  if $[-c_{\text{back}}, c_{\text{front}}] \subseteq [i_1, i_2]$ **then**
    return Content_Back + Content_Front // case 5
  end if
  if $i_2 \in [-c_{\text{back}}, 0]$ **then**
    return TriangContentBack(i_2) // case 3
  end if
  if $i_1 \in [0, c_{\text{front}}]$ **then**
    return TriangContentFront(i_1) // case 7
  end if
  if $i_2 \in [0, c_{\text{front}}]$ **then**
    return Content_Back + Content_Front − TriangContentFront(i_2) // case 4
  end if
  if $i_1 \in [-c_{\text{back}}, 0]$ **then**
    return Content_Front + Content_Back − TriangContentBack(i_1) // case 6
  end if
end function

Algorithm 5 CTriangle::GetMassL(x_beta, from, to)

```
1:  function GetMassL(x_β, from, to)
2:      mass ← 0
3:      i_1 ← from − x_β
4:      i_2 ← to − x_β
5:      if [−c_front, c_back] ∩ [i_1, i_2] = ∅ then
6:          return 0 // case 1, 2
7:      end if
8:      if [−c_front, c_back] ⊆ [i_1, i_2] then
9:          return Content_Front + Content_Back // case 5
10:     end if
11:     if i_2 ∈ [−c_front, 0] then
12:         return TriangContentFront(i_2) // case 3
13:     end if
14:     if i_1 ∈ [0, c_back] then
15:         return TriangContentBack(i_1) // case 7
16:     end if
17:     if i_2 ∈ [0, c_back] then
18:         return Content_Front + Content_Back − TriangContentBack(i_2) // case 4
19:     end if
20:     if i_1 ∈ [−c_front, 0] then
21:         return Content_Back + Content_Front − TriangContentFront(i_1) // case 6
22:     end if
23: end function
```

in the simulation running in discrete timesteps denoted as t_n, for which applies

$$t_n = t_{n-1} + h\,,$$

where h denotes the discretization parameter.

At each timestep, all sections of the model are updated. After the update, inactive agents with the arrival time occurring in the current timestep are passed to the initial transport section, and agents leaving their current sections are transferred to the subsequent section, as displayed in Figure 5.4. After passing all sections, each agent logs the final information from its trajectory and disappear.

The update algorithm of the transport section is implemented by Algorithm 6, where the parameter t_n represents current simulation time. Moreover, the function implements equations (4.3), (4.4), (4.5), (4.6), (4.7), (1.14), and (4.9). Functions GetMassB, and GetMassL represent calculation of the mass of agent $\beta \in$ B, $\beta \in$ L respectively, denoted as r_β in Section 4.2.1, in a surroundings given by parameters $from$ and to.

The update algorithm of the bottleneck section, described in Algorithm 7, operates with the parameter t_n representing the discrete time of the simulation. The algorithm progresses sequentially through all updates scheduled until time t_n. Furthermore, it records the order of agents based on their next update time. It generates updates for the agent that just completed its update and for the first agent in the queue following the movement of the preceding agent onto the lattice.

5.3 Use of the Simulation Tool

It is worth showing the directory structure of the project in Figure 5.6. File `Makefile` comprehends information for utility `CMake`. The command `make` compiles the `C++` source code located in

Algorithm 6 CTransport::Step(t_n)

procedure STEP(t_n)
  $speed \leftarrow$ Empty Map
  for each $\alpha \in$ B **do**
    $mass \leftarrow \sum_{\beta \in \mathrm{B} \setminus \{\alpha\}}$ GetMassB(x_β, $x_\alpha + d_1^{\mathrm{dir}}$, $x_\alpha + d_2^{\mathrm{dir}}$) // (4.3)
    $mass$ += $\sum_{\beta \in \mathrm{B} \setminus \{\alpha\}}$ GetMassL(x_β, $x_\alpha + d_1^{\mathrm{opp}}$, $x_\alpha + d_2^{\mathrm{opp}}$) // (4.4)
    $speed[ped] \leftarrow$ FD($mass$) // (4.9)
  end for
  for each $\alpha \in$ L **do**
    $mass \leftarrow \sum_{\beta \in \mathrm{B} \setminus \{\alpha\}}$ GetMassL(x_β, $x_\alpha - d_1^{\mathrm{dir}}$, $x_\alpha - d_2^{\mathrm{dir}}$) // (4.5)
    $mass$ += $\sum_{\beta \in \mathrm{B} \setminus \{\alpha\}}$ GetMassB(x_β, $x_\alpha - d_1^{\mathrm{opp}}$, $x_\alpha - d_2^{\mathrm{opp}}$) // (4.6)
    $speed[ped] \leftarrow$ FD($mass$) // (4.9)
  end for
  for each $\alpha \in$ A **do**
    $x_\alpha \leftarrow x_\alpha + v_\alpha \cdot speed[ped] \cdot$ Timestep // (1.14)
  end for
end procedure

Algorithm 7 CBottleneck::Step(t_n)

procedure STEP(t_n)
  while not IsEmpty(Update_Order) **do**
    $\alpha \leftarrow$ Front(Update_Order)
    if NextUpdate(α) $> t_n$ **then**
      break
    end if
    Pop(Update_Order)
    TryMove(α)
    if IsEmpty(SubsequentCell(α)) **then**
      MoveForward(α)
      AssignUpdateTime(α)
      if not IsEmpty(Queue) **and not** HasNextUpdate(Front(Queue)) **then**
        AssignUpdateTime(Front(Queue))
      end if
    else
      AssignUpdateTime(α)
    end if
  end while
end procedure

the directory `src`, and creates the executable file `simulation`. The executable has 3 parameters, that are further specified in the following list,

1. number of independent simulations run,
2. path to parameter file,
3. signal to log trajectories from transport section.

An example of an executing simulation is shown in the following bash command,

```
./simulation 1 params/full_simulation 1
```

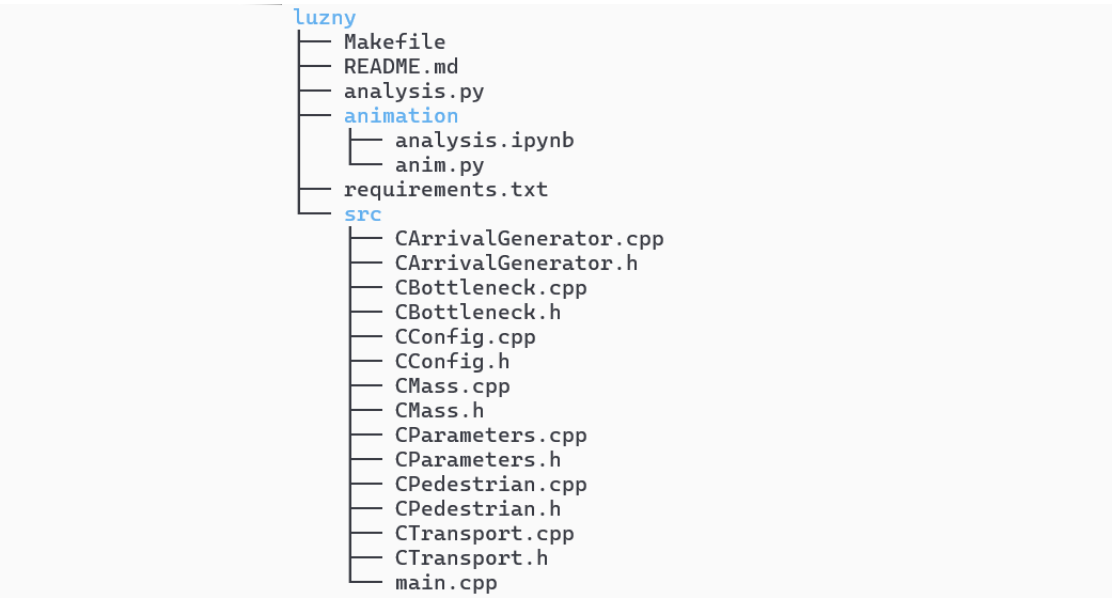

Figure 5.6 Directory structure within the thesis project

To be complete, the directory structure reveals `Python` script `analysis.py`, which is used to retrieve outputs analysed in Section 6.2, and `animation/anim.py`, which creates the animation presented in Figure 5.1. Morover, the directory `animation/` contains a `Jupyter` notebook `analysis.ipynb`, which presents validation of kernels, discussed in Section 6.1.

5.3.1 Entering Parameters

Initially, simulation parameters, discussed in the previous section, are read from a parameter file. The parameter file is a text file with the value of each parameter on one line. The first line of the parameter file is a path to log directory, where the output of the simulation is saved. It is followed by the values of parameters that are presented in Table 5.1. The order of the parameters is the same as displayed in the table.

All parameters are expected to be floating point numbers, except the log directory, which is expected to be text string. To be complete, a value that determines which kernels are run in the simulations follow. For each set of arrivals, an independent simulation is conducted for each selected kernel. The value is expected to be integer in range $0 - 4$. The following list explains the behavior for each value. The *no mass kernel* represents free flow.

- 0 – *Diraq kernel*, both *triangular kernels deflected foward and backwards*, and *no mass kernel*,

- 1 – only *Diraq kernel*,
- 2 – only *triangular kernel deflected forward*,
- 3 – only *triangular kernel deflected backwards*,
- 4 – only *no mass kernel*,

To be complete, triangular kernel deflected backwards is a reversal of Equation (4.14), and Equation (4.13).

A format of such a file is given by the following example,

```
log/log_dir/ # [string] path
0.1 # [double] timestep
0.75 # [double] body_width
1.2 # [double] speed_median
0.26 # [double] speed_std
0.45 # [double] speed_trim
4 # [double] R_crit
15 # [double] R_max
0 # [double] d1_dir
5 # [double] d2_dir
0 # [double] d1_op
5 # [double] d2_op
0.75 # [double] c_back
1.95 # [double] c_front
150 # [double] distance fd1
0 # [double] distance tasep
0 # [double] distance fd2
1500 # [double] time_intervalB
1500 # [double] time_intervalL
0.2 # [double] lambdaB
0.2 # [double] lambdaL
0 # [enum] kernels
```

In addition, the non-homogeneous arrivals are run when the parameters `lambdaB` and `lambdaL` are negative. Then the meaning of these parameters changes to the maximum arrival intensity on the given time interval and the probability function, discussed in Equation (1.2), is set on the additional lines of the parameter file as the coefficients of the polynomial function in the descending order. The format is given by the following example, where the polynomial $p_{\mathrm{B}} = 0.3x^4 - 0.4x^3 + 0.7x^2 + 0.9x + 0$.

```
0.3 -0.4 0.7 0.9 0
1 2 -1 0 2
```

It is noteworthy that the tool is able to run the simulation in multiple modes. In addition to run simulation with all sections of the model, the implementation enables to run only transport section or place not-moving artificial agents in the transport model. However, the project is not in final version and these features may will undergo major modifications in the subsequent work.

5.3.2 Outputs Files

The outputs of simulation are located in the directory determined by the first parameter of a parameter file. Due to the option of multiple iterations of the particular parameter configuration, each iteration is saved in the individual folder, named by the order of iteration, counted from 0.

For example, if the executable accepted 3 as iteration count, in the log directory will be folders 0, 1, and 2, containing output files for the corresponding runs of the simulation.

Each selected kernel has its own output files. The prefix of the log file name varies based on the kernel. The following list displays prefix for each kernel,

- Diraq kernel – `diraq`,
- triangular kernel deflected forward – `triang1`,
- triangular kernel deflected backwards – `triang2`.

A total of six output files are generated for each simulation run. Based on the content, each log file type has individual suffix separated from the prefix by a dot. An example of log file names for triangular kernel deflected forward is shown in the following text,

```
triang1.bottleneckB
triang1.bottleneckL
triang1.transportB
triang1.transportL
triang1.transportB_end
triang1.transportL_end
```

The files with the suffix `bottleneckB` or `bottleneckL` contain, on each line, the state of the bottleneck section at the time when the section undergoes an update of agents. Each line follows the format `timestamp:queue size:`, followed by a sequence of records for each present agent separated by a space. An example of two lines from such a log file is provided below,

```
179.54:0:44,13 49,12
179.59:0:45,13 49,12
```

In this example, at a simulation time of 179.54 seconds, there are 0 agents present in the queue, and the agents with IDs 13 and 12 are situated in this section at positions 45 and 49, respectively. The position of the cell is counted from the Březník side for both instances of the bottleneck section.

Furthermore, in the files with the suffix `transportB` or `transportL`, counts of active agents are recorded for each stream on one line, for every timestep of the simulation. Additionally, when the executable receives a signal to log trajectories, each line is followed by a sequence consisting of the agent's ID, current speed, and position, with each value separated by a comma. Each log entry is separated by a space. An example of such a log file is provided below,

```
2 11,1.19,3.21 10,1.13,10.04
1 0,1.43,22.23
```

In this example, there are 2 active B-type agents, with the first agent having an ID of 11, a current speed of 1.19 m/s, and a position of 3.21 m. Additionally, there is 1 active L-type agent with an ID of 0, a speed of 1.43 m/s, and a position of 22.23 m.

The files with the suffix `transportB_end` or `transportL_end` contain travel time information for agents that have completed their trajectory. Each line consists of a sequence of optimal velocity, time of passing the last section of the model, and the arrival time, separated by commas. Each log entry is separated by a space. An example of such a log file is provided below:

```
-1.43,107.70,0.09 -1.64,109.60,11.80 -1.64,109.60,11.80
```

In this example, 3 agents completed their trajectory on the first transport section of the model, indicating they are of L-type. The optimal velocity of the first agent is -1.43 m/s, which completed the trajectory at the simulation time 107.70 s, and reached the trail at time 0.09 s.

Chapter 6

Analysis of Simulation Output

The simulation tool presented in the previous chapter primarily serves to perform experiments. This thesis introduces two variants of experiments. Validation of kernel implementation is presented in Section 6.1, and development of macroscopic pedestrian flow characteristics at increasing arrival intensity is discussed in Section 6.2.

It is important to note that all experiments presented in this chapter are conducted on simulations of the transport section only. The reason is that neither of the experiments requires the simple bottleneck section, which is introduced in this thesis.

6.1 Validation of Kernel Implementation

The functionality of the simulation tool is partially validated by the animation shown in Figure 5.1. However, the most error-prone section of the implementation is the kernel calculations. The implementations of Algorithm 4, and Algorithm 5, which are presented in the previous chapter, are validated in the following experiments.

The experiments involve three kernels, namely

- Diraq kernel function, introduced in Equation (2.9),
- triangular kernel function deflected forward, presented in equations (4.13), (4.14),
- and triangular kernel function deflected backwards, discussed in Section 5.3.1.

The experiments are designed to verify the velocity profile of the agents during various scenarios. Each kernel is tested in the following set of 5 scenarios. The first scenario from the list is illustrated in Figure 6.1.

1. 1 B-type agent walks past 1 not-moving L-type agent,
2. 1 L-type agent walks past 1 not-moving B-type agent,
3. 1 B-type agent walks past 1 not-moving B-type agent,
4. 1 L-type agent walks past 1 not-moving L-type agent,
5. 1 B-type agent, 1 L-type agent, walk past each other.

In each experiment, the speed of the moving agents is set to 1 m/s and the length of the transport section is set to 20 m. Not-moving agents are located at position 10 m. The parameters of the velocity-governing function, presented previously in Equation (4.10), are modified to

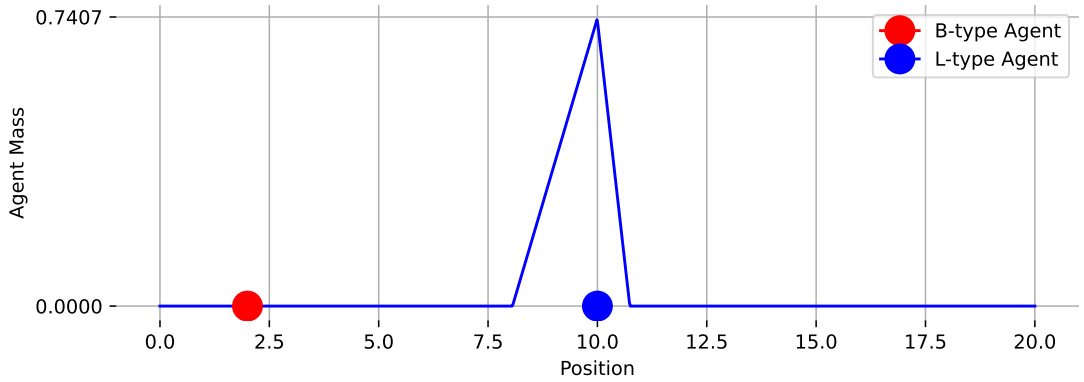

Figure 6.1 Illustration of the scenario 1 B-type agent walks past 1 not-moving L-type agent with triangular kernel function deflected forwards

capture the interaction between agents. The parameter $R_{\max}$ is set to value 2 in all experiments. The remaining simulation parameters are based on the calibration discussed in Section 5.1.

In the first set of experiments, the critical agent mass is set to $R_{\text{crit}} = 0$, which captures the reduction of speed in the velocity profile for as long as the mass of the other agent overlays with the perceived surroundings. Figure A.1, Figure A.2, and Figure A.3 in Appendix A.1 display the velocity profile of the moving agents during the 5 scenarios presented previously, for Diraq kernel, triangular kernel deflected forward, and triangular kernel deflected backwards respectively.

In the second set of experiments, the critical agent mass is set to $R_{\text{crit}} = 0.5$ to capture a milder reaction of the agent. Figure A.4, Figure A.5, and Figure A.6 in A.1 display the velocity profile for the scenarios $1, 2, 5$ from the enumerate list presented previously.

In the experiments with the Diraq kernel, a sudden reaction to the agent mass causes the velocity profile of agents to be piecewise constant function on two intervals. The interval, in which the other agent is located in the perceived surroundings, and the interval, in which the other agent is out of the perceived surroundings.

The velocity profile of B-type agent walking past not-moving L-type agent is displayed in Figure 6.2. The perceived surroundings $[x_\alpha(t) + d_1^{\text{opp}}, x_\alpha(t) + d_1^{\text{opp}}]$ of the B-type agent α includes the position 10 m, at which is the L-type agent located, for $x_\alpha(t) \in [5, 10]$, which matches with the the results from the simulation. The results presented in Figure A.4 in Appendix A.1 can be validated similarly.

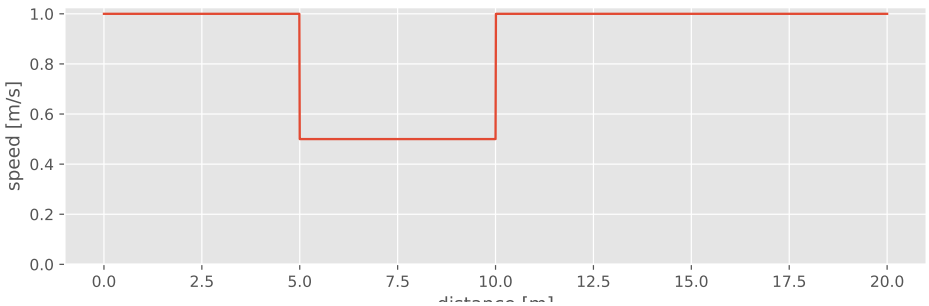

Figure 6.2 Velocity profile of the moving agent in the scenario 1 B-type walks past 1 not-moving L-type agent with Diraq kernel

The following calculation is limited to a situation, in which the B-type agent, with parameter

$R_{crit} = 0$, walks past static L-type agent. The individual mass generated by the L-type agent is the triangular kernel deflected forward. To be complete, the discussed case results are depicted in Figure 6.3 on the left.

In this scenario, the speed reduction in the velocity profile is determined by 5 different functions, based on the part of the triangular kernel that overlays with the perceived surroundings. The possible overlays are illustrated previously in Figure 5.5. The position intervals of the agent, in which the speed reduction follows different function, are presented in the following list. For each interval, the color used in the derived velocity profile in Figure 6.3 is specified in brackets. As a point of reference, the values of kernel parameters are $c_{\text{front}} = 1.95$ m, and $c_{\text{back}} = 0.75$ m.

1. the front part of the kernel is partially perceived in position range $3.05 - 5$ m (red),
2. the front part of the kernel is completely perceived and the back part of the kernel is partially perceived in position range $5 - 5.75$ m (blue),
3. both the front and the back part of the kernel are perceived completely in position range $5.75 - 8.05$ m (yellow),
4. the front part of the kernel is partially perceived and the back part of the kernel is completely perceived in position range $8.05 - 10$ m (green),
5. the back part of the kernel is partially perceived in position range $10 - 10.75$ m (orange).

Furthermore, the agent mass in the perceived surroundings is determined solely by the formula presented previously in Equation (4.4). On each interval from the previous enumerated list, the formula is represented by an individual function. For this scenario, the functions that describe the agent mass in the perceived surroundings on each interval are presented in the following equations (6.1) – (6.5). For a more comprehensive understanding, the area of the back part of the kernel is $\frac{5}{18}$, the area of the front part of the kernel is $\frac{13}{18}$, and the normalizing constant C is $\frac{2}{2.7}$.

$$R = \int_{8.05}^{x+5} \frac{2}{2.7} \cdot \left(1 - \frac{10 - y}{1.95}\right) dy + 0 = 1.76686 - 1.15859x + 0.189934x^2 \qquad \text{for } x \in [3.05, 5] \,, \tag{6.1}$$

$$R = \frac{13}{18} + \int_{10}^{x+5} \frac{2}{2.7} \cdot \left(1 - \frac{y - 10}{0.75}\right) dy = -15.3272 + 5.67901x - 0.493827x^2 \qquad \text{for } x \in [5, 5.75] \,, \tag{6.2}$$

$$R = \frac{13}{18} + \frac{5}{18} = 1 \qquad \text{for } x \in [5.75, 8.05] \,, \tag{6.3}$$

$$R = \int_{x+0}^{10} \frac{2}{2.7} \cdot \left(1 - \frac{10 - y}{1.95}\right) dy + \frac{5}{18} = -11.3082 + 3.05793x - 0.189934x^2 \qquad \text{for } x \in [8.05, 10] \,, \tag{6.4}$$

$$R = 0 + \int_{x+0}^{10.75} \frac{2}{2.7} \cdot \left(1 - \frac{y - 10}{0.75}\right) dy = 57.0679 - 10.6173x + 0.493827x^2 \qquad \text{for } x \in [10, 10.75] \,. \tag{6.5}$$

The velocity is determined by the formula presented previously in Equation (4.10). Once the values $v_\alpha = 1$, $R_{crit} = 0$ and $R_{max} = 2$ have been assigned to the formula parameters, the obtained velocity-mass relation for this scenario is presented in the following equation.

$$v(R) = 1 - \frac{R}{2} \,. \tag{6.6}$$

Finally, substitution of the agent mass R in Equation (6.6) with the corresponding function from equations (6.1) – (6.5) retrieves the function of the velocity profile on each interval, i.e.

$$v(\mathrm{R}) = 1 + \frac{1}{2}\left(-1.76686 + 1.15859x - 0.189934x^2\right) \qquad \text{for } x \in [3.05, 5]\ , \tag{6.7}$$

$$v(\mathrm{R}) = 1 + \frac{1}{2}\left(15.3272 - 5.67901x + 0.493827x^2\right) \qquad \text{for } x \in [5, 5.75]\ , \tag{6.8}$$

$$v(\mathrm{R}) = 1 - 0.5 \qquad \text{for } x \in [5.75, 8.05]\ , \tag{6.9}$$

$$v(\mathrm{R}) = 1 + \frac{1}{2}\left(-26.6785 + 4.53941x - 0.189934x^2\right) \qquad \text{for } x \in [8.05, 10]\ , \tag{6.10}$$

$$v(\mathrm{R}) = 1 + \frac{1}{2}\left(41.142 - 9.1358x + 0.493827x^2\right) \qquad \text{for } x \in [10, 10.75]\ . \tag{6.11}$$

The derived velocity profile is compared with the simulation velocity profile in Figure 6.3. A match of the simulation velocity profile with the derived velocity profile confirms the correctness of the implementation of the triangular kernels. The velocity profiles for the remaining experiments presented in Figure A.2, Figure A.3, Figure A.5 and Figure A.6 in Appendix A.1 can be derived similarly.

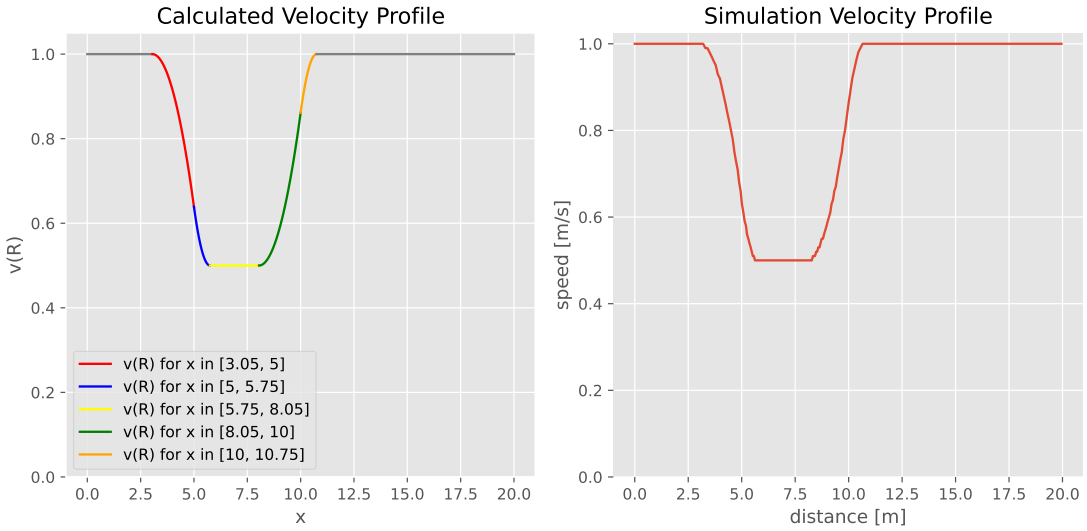

Figure 6.3 Comparison of the calculated velocity profile and the velocity profile from the simulation of the scenario 1 B-type walks past 1 not-moving L-type agent with triangular kernel deflected forward

6.2 Experiments with Arrival Intensity

The arrival intensity is expected to have the greatest impact on the capacity of the trail. The experiments presented in this section have several purposes. Mainly, all experiments aim to detect the intensity of arrivals, at which the pedestrian flow stops to be smooth and jams occur. Further, the impact of heterogeneity in velocities and heterogeneity in arrival intensity on the pedestrian flow is observed. Additionally, the experiments serve as a comparison of behavior for the various implemented kernels. Moreover, the results are compared with the empirical findings presented previously in Chapter 3.

In the experiments presented in this section, the average arrival intensity is identical for both ends of the trail. In the simulation results presented in this section, the intensity value applied

to the x-axis is the average intensity of arrivals to one boundary of the trail. To be complete, the total average intensity of the arrivals is twice the average arrival intensity to one boundary of the trail. It is possible to conduct advanced experiments with unbalanced bidirectional flow in the subsequent work.

Below are presented two groups of experiments. The first group of experiments, presented in Section 6.2.2, performs simulations with homogeneous arrival intensity for three scenarios of speed distribution of agents. The second group of experiments, presented in Section 6.2.3, performs simulations with non-homogeneous arrival intensity for three arrival intensity scenarios.

All experiments share the values of the parameters presented in the following Table 6.1. Except for the lengths of the bottleneck section and the transport section 2, which are omitted, the parameter values are selected based on the calibration presented previously in Section 5.1. In the experiments with non-homogeneous arrival intensity, the speed distribution parameter $\text{speed}_{\text{trim}}$, discussed previously in Section 5.1, is set to the calibrated value 0.45 m/s.

Parameter Name	Variable	Selected Value	Units
timestep	h	0.1	s
body width	w_{cell}	0.75	m
speed median	$\text{speed}_{\text{med}}$	1.2	m/s
speed deviation	$\text{speed}_{\text{std}}$	0.26	m/s
critical mass	R_{crit}	15/3.7	mass of pedestrians
maximum mass	R_{max}	15	mass of pedestrians
d1 dir	d_1^{dir}	0	m
d2 dir	d_2^{dir}	5	m
d1 opp	d_1^{opp}	0	m
d2 opp	d_2^{opp}	5	m
c back	c_{back}	0.75	m
c front	c_{front}	1.95	m
distance transport1	$\lvert x_{\text{end, 1}} - x_{\text{start, 1}} \rvert$	150	m
distance bottleneck	$\lvert x_{\text{start, 2}} - x_{\text{end, 1}} \rvert$	0	m
distance transport2	$\lvert x_{\text{end, 2}} - x_{\text{start, 2}} \rvert$	0	m
arrival period B	$t_{\text{max}}^{\text{B}}$	1500	s
arrival period L	$t_{\text{max}}^{\text{L}}$	1500	s

Table 6.1 Parameters with values common to all experiments

The simulation starts at the simulation time 0 s and ends after the departure of the last active agent. In case of total stoppages, the simulations are ended at the simulation time 2000 s. The end of each simulation is further represented by the variable t_{stop}, which is given by Equation (6.12). The variables $t_{\text{last}}^{\text{B}}$, $t_{\text{last}}^{\text{L}}$ refer to the last departure of B-type agent, L-type agent respectively.

$$t_{\text{stop}} = \min\{\max\{t_{\text{last}}^{\text{B}}, t_{\text{last}}^{\text{L}}\}, 2000\}\,. \tag{6.12}$$

6.2.1 Analysed Characteristics

The experiments presented in this section analyse two macroscopic characteristics of pedestrian flow, namely

1. average count of active pedestrians,
2. average velocity of agents.

The set of agents and the time period that are included in the calculations differs for the experiments with homogeneous and non-homogeneous arrival intensity. Therefore, the sets are introduced before the introduction of the characteristics.

To prevent initial arrivals on the trail from distorting the results in the experiments with the homogeneous arrival intensity, a **stationary state** is established. The motivation is that after the departure of agents from both walking directions, the pedestrian flow with the homogeneous arrival intensity is already established.

The set of agents, denoted as Astat, whose whole trajectories belong to the stationary state, is defined by Equation (6.13). The discrete time steps Tstat corresponding to the stationary state are determined by Equation (6.14). The variables $t^{\mathrm{B}}_{\mathrm{earl}}$, $t^{\mathrm{L}}_{\mathrm{earl}}$ refer to the earliest departure of B-type agent, L-type agent respectively. The variable t_{stop} is established above in Equation (6.12).

$$\mathrm{A}_{\mathrm{stat}} = \{\alpha \in \mathrm{A} \,|\, t^{\alpha}_{\mathrm{arr}} \geq \max\{t^{\mathrm{B}}_{\mathrm{earl}}, t^{\mathrm{L}}_{\mathrm{earl}}\}\}\,, \tag{6.13}$$

$$\mathrm{T}_{\mathrm{stat}} = \{t_n \,|\, n \in \mathbb{N} \wedge t_n \in \left[\max\{t^{\mathrm{B}}_{\mathrm{earl}}, t^{\mathrm{L}}_{\mathrm{earl}}\}, t_{\mathrm{stop}}\right]\}\,. \tag{6.14}$$

An attempt was made to impose a tighter condition by establishing an upper limit of the duration of the stationary state. The upper limit is not established due to the fact that the agents, who completely jammed and therefore never reach the end of the trail, would not belong to the stationary state.

The calculation of the characteristics for the experiments with non-homogeneous arrival intensity includes all agents in the simulation. Further, the time period T ranges from the simulation time 0 s to the end of the simulation t_{stop}.

The calculation of **average count of active pedestrians**, denoted as $\bar{\mathrm{N}}$, is introduced in Equation (6.15). A' represents a set of agents and T' denotes a set of discrete simulation timesteps that are included in the calculation, as discussed above.

$$\bar{\mathrm{N}} = |T^{-1}| \cdot \sum_{t_n \in T} \left|\{\alpha \,|\, \alpha \in \mathrm{A} \wedge t_n \in \left[t^{\alpha}_{\mathrm{arr}}, t^{\alpha}_{\mathrm{dep}}\right]\}\right|\,. \tag{6.15}$$

The calculation of **average velocity of agents**, denoted as $\bar{v}$, is presented in Equation (6.17), where $\bar{v_\alpha}$ denotes the average speed of agent $\alpha \in \mathrm{A}_{\mathrm{stat}}$, given by Equation (6.16).

$$\bar{v_\alpha} = |x_{\mathrm{end}} - x_{\mathrm{start}}| \cdot (t^{\alpha}_{\mathrm{dep}} - t^{\alpha}_{\mathrm{arr}})^{-1}\,, \tag{6.16}$$

$$\bar{v} = |\mathrm{A}_{\mathrm{stat}}|^{-1} \cdot \sum_{\alpha \in \mathrm{A}_{\mathrm{stat}}} \bar{v}_\alpha\,. \tag{6.17}$$

Without interactions between agents, the average speed of agents is theoretically the mean value of the speed distribution, i.e. $\mathrm{speed}_{\mathrm{med}} = 1.2$ m/s, and the average count of active pedestrians at the average arrival intensity λ is given by the following formula, where $\bar{t}$ represents the average time of agents to walk the trail with length $|x_{\mathrm{end}} - x_{\mathrm{start}}|$, i.e.

$$\mathrm{E}\bar{\mathrm{N}} = (\lambda + \lambda) \cdot \bar{t} = 2 \cdot \lambda \cdot \frac{|x_{\mathrm{end}} - x_{\mathrm{start}}|}{\mathrm{speed}_{\mathrm{med}}} = 250 \cdot \lambda\,. \tag{6.18}$$

Figure A.13, Figure A.14 in Appendix A.2 and Figure A.21, Figure A.22 in Appendix A.3 show the experiment results, where the agent mass is ignored by the agents. As expected, the theoretical values presented above match the results.

At the agent mass in the perceived surroundings of the agent equal to $\mathrm{R}_{\mathrm{max}}$, the agent completely stops. Therefore, the maximum count of active agents in the simulation step is limited by the following formula, where $\dfrac{\mathrm{R}_{\mathrm{max}}}{|d_2 - d_1|}$ represents the maximum count of agents on a one-meter long section and $|x_{\mathrm{end}} - x_{\mathrm{start}}|$ represents the length of the trail, i.e.

$$\mathrm{N}_{\mathrm{max}} = \frac{\mathrm{R}_{\mathrm{max}}}{|d_2 - d_1|} \cdot |x_{\mathrm{end}} - x_{\mathrm{start}}|\,. \tag{6.19}$$

Equations (6.18), and (6.19) together form an upper estimate of the maximum arrival intensity, at which each pedestrian is guaranteed to stop, given by Equation (6.20). In case of the experiments presented in this section, this corresponds to the intensity of 1.8 ped/s for each direction, i.e. flow rate equal to 216 ped/min. This value is comparable with the recommendations presented in Table 3.4, where the flow rate on the walkway with width is recommended to be in range $10 - 200$ ped/min.

$$\lambda_{\max} = \frac{\mathrm{N}_{\max}}{2} \cdot \frac{\mathrm{speed}_{\mathrm{med}}}{|x_{\mathrm{end}} - x_{\mathrm{start}}|} = \frac{\mathrm{R}_{\max}}{2} \cdot \frac{\mathrm{speed}_{\mathrm{med}}}{|d_2 - d_1|}\,, \tag{6.20}$$

6.2.2 Experiments with Homogeneous Arrival Intensity

This section presents three experiments with homogeneous arrival intensity, each with different agent speed distribution. The motivation for these experiments is the assumption that increasing heterogeneity in the speed of agents leads to jams at lower arrival intensities.

The speed distribution scenarios are presented in Table 6.2. The speed distribution that is is selected in the parameter calibration previously in Section 5.1 is further denoted as $\mathrm{S}_{\mathrm{dist}}$.

Table 6.2 Description of the speed distribution scenarios in the experiments with homogeneous arrival intensity

Scenario Name	Description
Speed Distribution Scenario 1	$v_\alpha = \mathrm{speed}_{\mathrm{med}}$ for $\alpha \in \mathrm{A}$
Speed Distribution Scenario 2	$v_\alpha = \mathrm{speed}_{\mathrm{med}}$ for $\alpha \in \mathrm{B} \wedge v_\beta$ follows $\mathrm{S}_{\mathrm{dist}}$ for $\alpha \in \mathrm{L}$
Speed Distribution Scenario 3	v_α follows $\mathrm{S}_{\mathrm{dist}}$ for $\alpha \in \mathrm{A}$

The experiments were performed with the values of the arrival intensity between 0.2 and 0.7 ped/s. To achieve revealing results, a total of 10 iterations are performed for each intensity in the experiments. In each iteration, the instances of agents were the same for all kernels.

The experiment results for the speed distribution scenarios 1 and 3 with the triangular kernel deflected forward are shown in Figure 6.4. The results for the remaining kernels and speed scenarios are displayed in Appendix A.2. The results of the 10 iterations for each arrival intensity are represented by a boxplot. A rapid increment of the average pedestrian count, rapid decrement of the average agent speed respectively, indicate the agent stoppages. The theoretical values at no interaction between agents, discussed above in Section 6.2.1, are plotted for comparison.

Regarding the value, at which the speed reduction occurs, the arrival intensity is around 0.3 ped/s. As shown in Table 6.3, the intensity at which jams occur differs based on the speed heterogeneity. At heterogeneous speed, the arrival intensity is approximately 0.05 less than for homogeneous speed of agents. This supports the assumption that heterogeneity in speed leads to stoppages at lower arrival intensity. Moreover, the value 0.65 ped/s, which is equal to the flow rate 78 ped/min for both walk direction, is comparable with the recommendations presented previously in Table 3.4, where it is recommended for the walkaway with flow rate 100 ped/min to be 2.5 m wide.

Table 6.3 Overview of the results of the experiments with the homogeneous arrival intensity and the triangular kernel deflected forward

Scenario	Deterioration in Flow	Stoppages
Speed Distribution Scenario 1	0.6 ped/s	0.7 ped/s
Speed Distribution Scenario 2	0.6 ped/s	0.65 ped/s
Speed Distribution Scenario 3	0.55 ped/s	0.65 ped/s

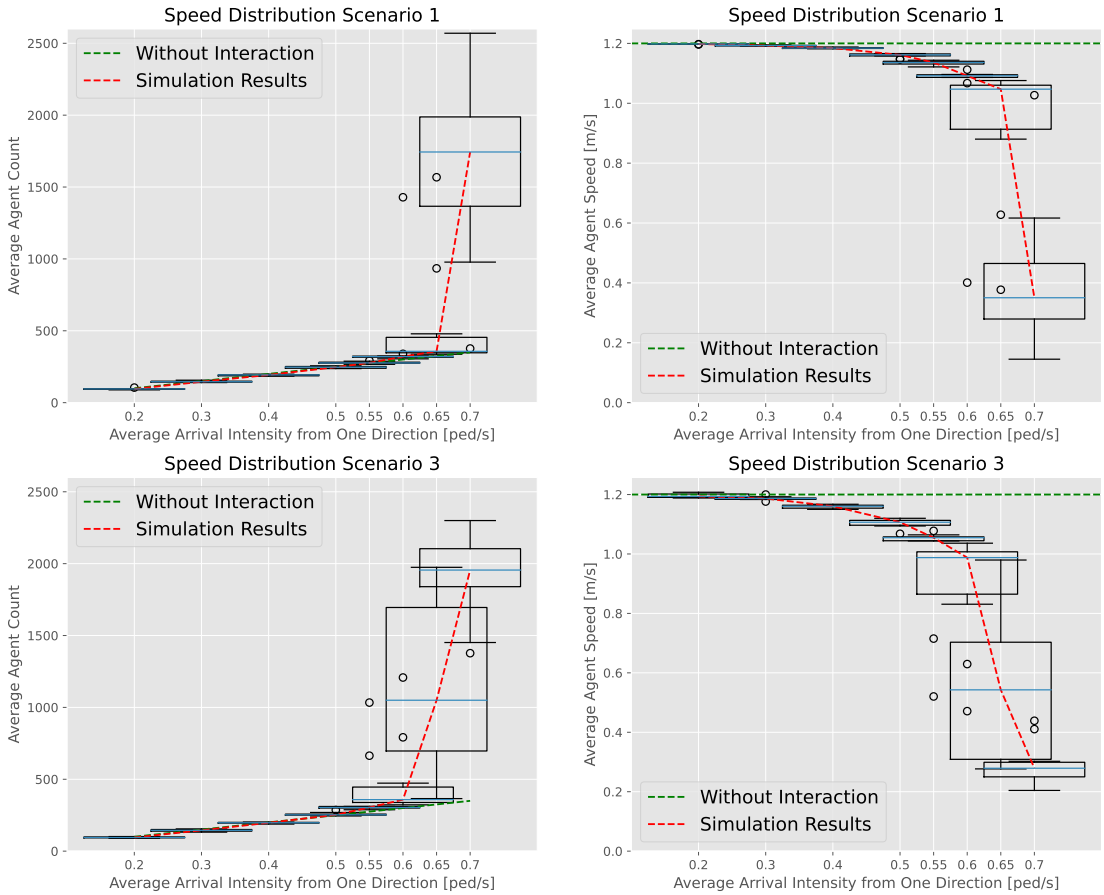

Figure 6.4 Results of experiments with homogeneous arrival intensity for homogeneous and heterogeneous speed distributions

6.2.3 Experiments with Non-homogeneous Arrival Intensity

This section presents experiments with arrival intensity for three different non-homogeneous arrival scenarios that are presented below in Table 6.4. The generic shapes of the probability functions λ_1, λ_2 and λ_3 that represent the non-homogeneous arrival intensity are depicted in Figure 6.5.

For implementation purposes, the probability functions are polynomials. The coefficients of the generic polynomials λ_1, λ_2 and λ_3 are the solution of the following coupled linear equations, where a, b, c, d and e are coefficients of a polynomial in the descending order.

$$\left(\begin{array}{ccccc|ccc} \mathbf{e} & \mathbf{d} & \mathbf{c} & \mathbf{b} & \mathbf{a} & \lambda_\mathbf{1} & \lambda_\mathbf{2} & \lambda_\mathbf{3} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 375 & 375^2 & 375^3 & 375^4 & \frac{2}{3} & 1 & 0.4 \\ 1 & 750 & 750^2 & 750^3 & 750^4 & 1 & 0.8 & 0.8 \\ 1 & 1125 & 1125^2 & 1125^3 & 1125^4 & \frac{2}{3} & 0.4 & 1 \\ 1 & 1500 & 1500^2 & 1500^3 & 1500^4 & 0 & 0 & 0 \end{array}\right)$$

The polynomials that represents the non-homogeneous arrival intensity with the average arrival intensity λ are obtained by scaling the generic polynomials with value

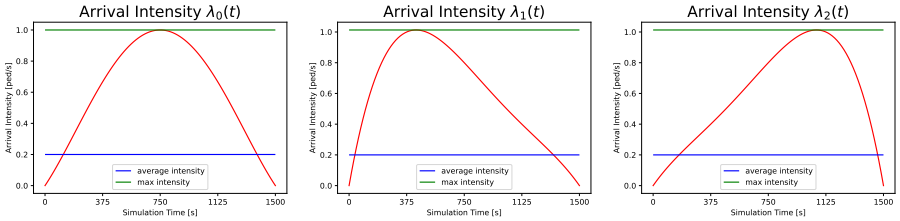

Figure 6.5 Illustration of the arrival intensities used in experiments with non-homogeneous arrival intensity

$$\lambda \cdot t_{\max} \cdot \left(\int_0^{t_{\max}} \lambda_i(t)\, dt \right)^{-1} \quad \text{for } i \in \{1, 2, 3\}\,.$$

Therefore, the average arrival intensity of the non-homogeneous arrival intensity represented by function $\lambda(t)$ is comparable with the homogeneous arrival intensity λ presented in the previous section, i.e.

$$\int_0^{t_{\max}} \lambda(t)\, dt = \lambda \cdot t_{\max}\,. \tag{6.21}$$

Table 6.4 Description of the arrival intensity scenarios in the experiments with non-homogeneous arrival intensity

Scenario Name	Arrival Intensity of B-type agents	Arrival Intensity of L-type agents
Arrival Intensity Scenario 1	λ_1	λ_1
Arrival Intensity Scenario 2	λ_2	λ_3
Arrival Intensity Scenario 3	λ_2	homogeneous arrival intensity λ

Due to the occurrence of earlier stoppages, the experiments were performed with the average arrival intensity ranging from 0.2 to 0.55 ped/s. Again, a total of 10 iterations are performed for each average intensity in the experiments and in each iteration, the instances of agents were the same for all kernels.

Similarly to the results presented in the previous section, the experiment results for the arrival intensity scenarios 1 and 2 with the triangular kernel deflected forward are shown in Figure 6.6. The results for the remaining kernels and speed scenarios are displayed in Appendix A.3.

In comparison to the experiments in the previous section, the deterioration in flow occurs at the average arrival intensity value of approximately 0.2 less than at homogeneous arrival intensity. The overview of the results is shown in Table 6.5

Further, the experiments reveal significant stoppages in the pedestrian flow at the average arrival intensity of 0.4 ped/s, which corresponds to the value of 48 ped/min. Based on the recommendations presented in Table 3.3 and Table 3.4, the flow rate at which stoppages occur should take the values around 100 ped/min. Therefore, the maximum arrival intensity coming from the simulation results is approximately two times less than recommended value. However, it is of the same order of magnitude which corroborates the reasonableness of the model. Based on the results, the arrival intensity scenario 1 has the earliest stoppages. The reason may be that the peak in arrival intensity is at the same time for both walking directions.

The comparison of all scenarios presented above for each kernel is shown in Figure 6.7 and Figure 6.8. Based on the figures, the experiments with non-homogeneous arrival intensity show significantly different results.

The comparison of various kernels is shown in Figure 6.9 and Figure 6.10. As expected, stoppages for Diraq kernel occur at lower arrival intensity. Although the difference between

results from various kernels is not significant, the two variants of triangular kernel show differences that can be targeted in the subsequent work.

Table 6.5 Overview of the results of the experiments with the non-homogeneous arrival intensity and the triangular kernel deflected forward

Scenario	Deterioration in Flow	Stoppages
Arrival Intensity Scenario 1	0.3 ped/s	0.4 ped/s
Arrival Intensity Scenario 2	0.4 ped/s	0.5 ped/s
Arrival Intensity Scenario 3	0.4 ped/s	0.5 ped/s

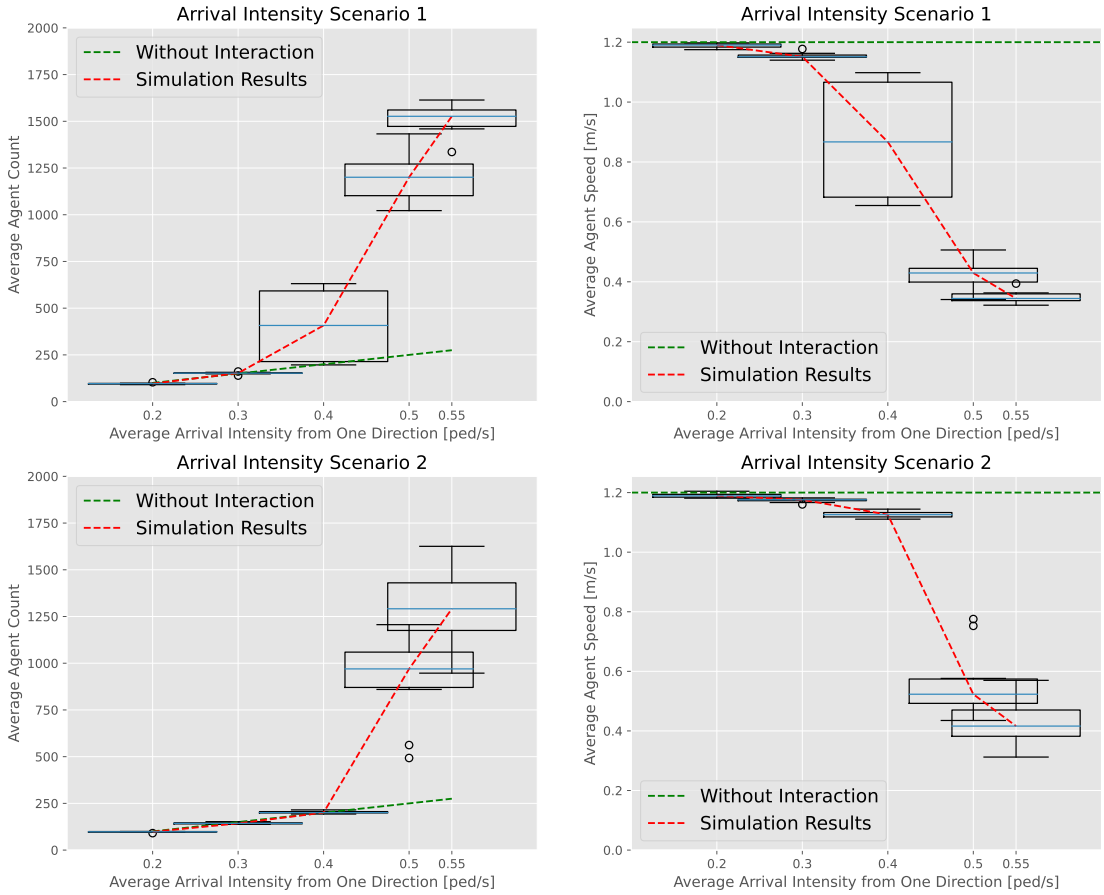

Figure 6.6 Results of experiments with non-homogeneous arrival intensity for 2 arrival intensity scenarios

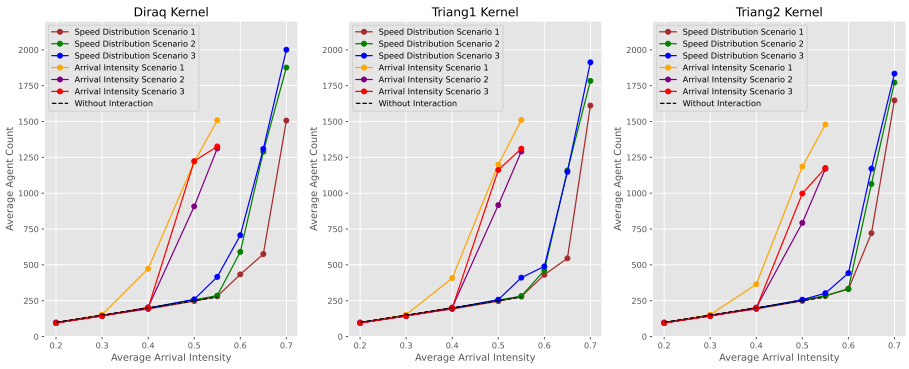

Figure 6.7 Comparison of average agent count results from all scenarios for each kernel

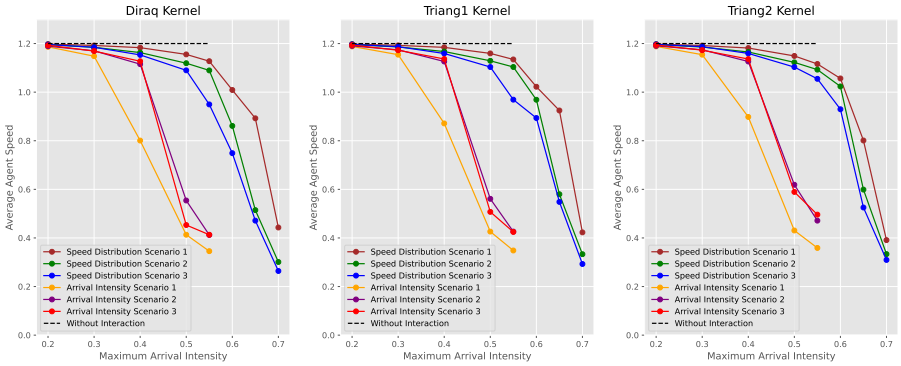

Figure 6.8 Comparison of average agent speed results from all scenarios for each kerne

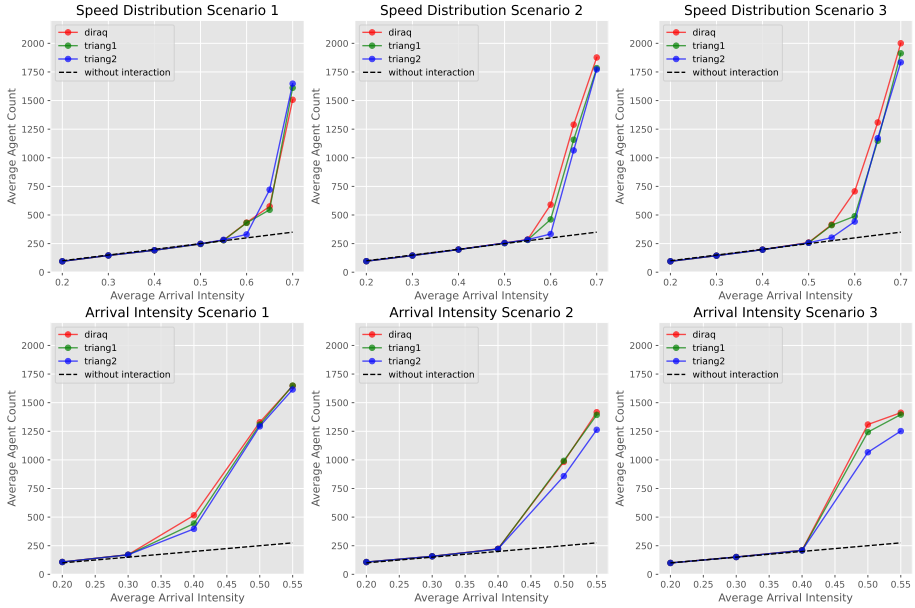

Figure 6.9 Comparison of kernels in agent count results for each scenario

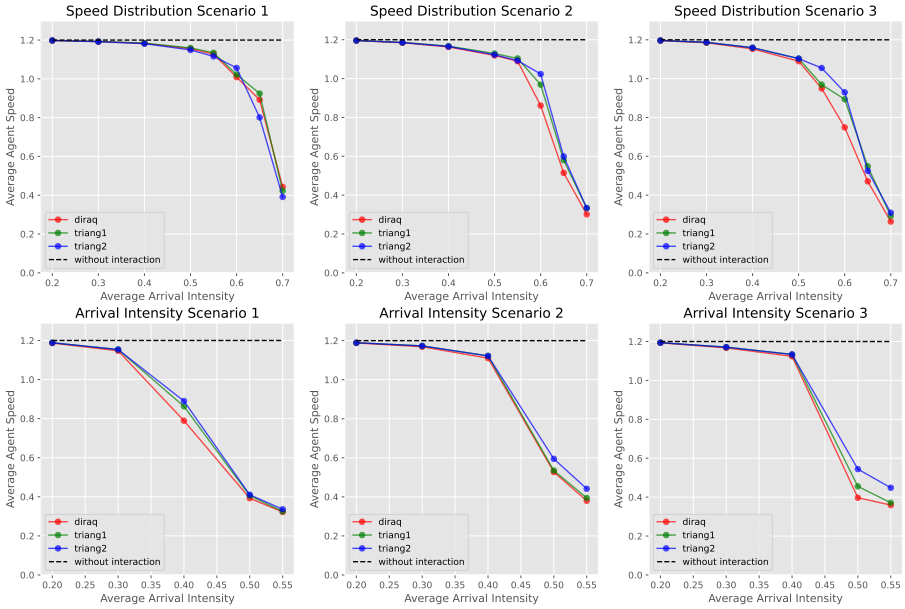

Figure 6.10 Comparison of kernels in agent speed results for each scenario

Chapter 7
Conclusion

The thesis successfully introduced an agent-based model that represents pedestrian flow in the Luzen Valley trail. The model consists of three distinct mathematical models, each defined in Chapter 4. Random arrivals are implemented by means of a Poisson process, discussed in Section 1.3. The transport section, situated at both ends of the trail, employs a fundamental diagram-based model. In Section 4.2.1, a novel concept for interaction between agents is introduced, namely the agent mass in the perceived surroundings of the agent. The individual agent mass is represented by kernel estimates presented in the article [1]. This thesis focuses on the transport section. The bottleneck section is currently represented by very simple cellular automaton, which is discussed in Section 1.2. The subsequent work aims to replace this model with more complex one.

Furthermore, the simulation tool was developed to conduct experiments with the model. In addition to the simulation tool, a script was created to provide a basic animation of the trajectories from the simulation. The implementation of the model is discussed in Chapter 5. Data from Chapter 3, which provides valuable empirical findings and an overview of work related to pedestrian movement, served in Section 5.1 to calibrate model parameters.

One of the two experiments conducted is an additional validation of kernels implementation, presented in Section 6.1. The second set of experiments, discussed in Section 6.2, identified the average arrival intensity of 48 ped/s, at which jams occur in the model. The empirical findings presented in this thesis confirm the reasonableness of this value. Additionally, the experiments compared the behavior of various kernels, which exhibited a high degree of similarity. Furthermore, an assumption that heterogeneity in speed of agents and heterogeneity in arrival intensity leads to earlier stoppages was confirmed in the experiments.

The subsequent work aims to estimate the capacity of the trail. To this end, an extension of the simulation tool, introduced in the thesis, is required to enable various restriction models on the trail to be tested. In addition, further evaluation of the fundamental diagram-based approach is appropriate to be done. As previously stated, the bottleneck section will be replaced with a more complex model, such as an expert system, in order to reflect the complexity of the current system.

Appendix A
Experiment Results

A.1 Results of Kernel Validation Experiment

This section presents the experiment results related to the validation of the kernel implementation, which is presented in Section 5.2. The kernels included in the experiments are the Diraq kernel, the triangular kernel deflected forward and the triangular kernel deflected backwards. In the first set of the experiments, the critical mass of agents is set to 0, and in the second set of the experiments, the critical mass of agents is set to 0.5.

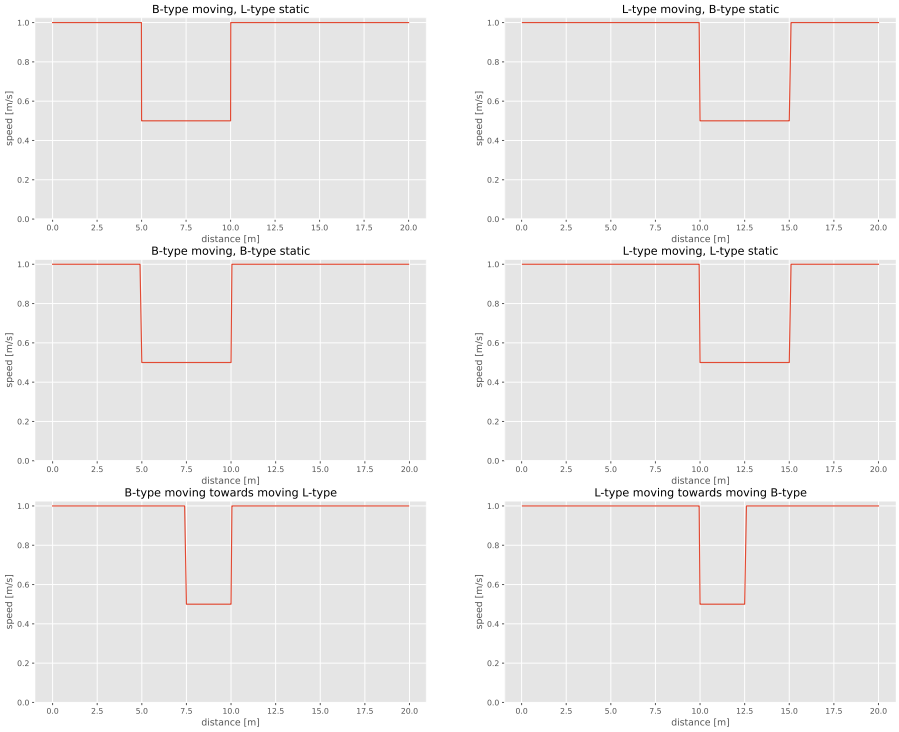

Figure A.1 Velocity profiles for 5 scenarios, Diraq kernel used as individual mass of agent, $R_{crit} = 0$

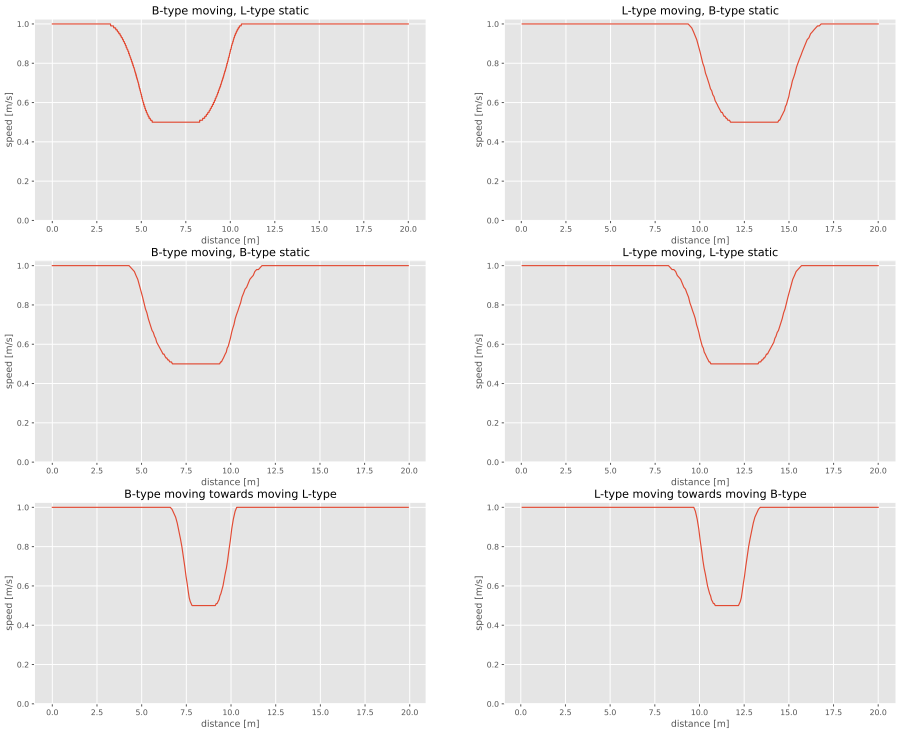

Figure A.2 Velocity profiles for 5 scenarios, triangular kernel deflected forward used as individual mass of agent, $R_{crit} = 0$

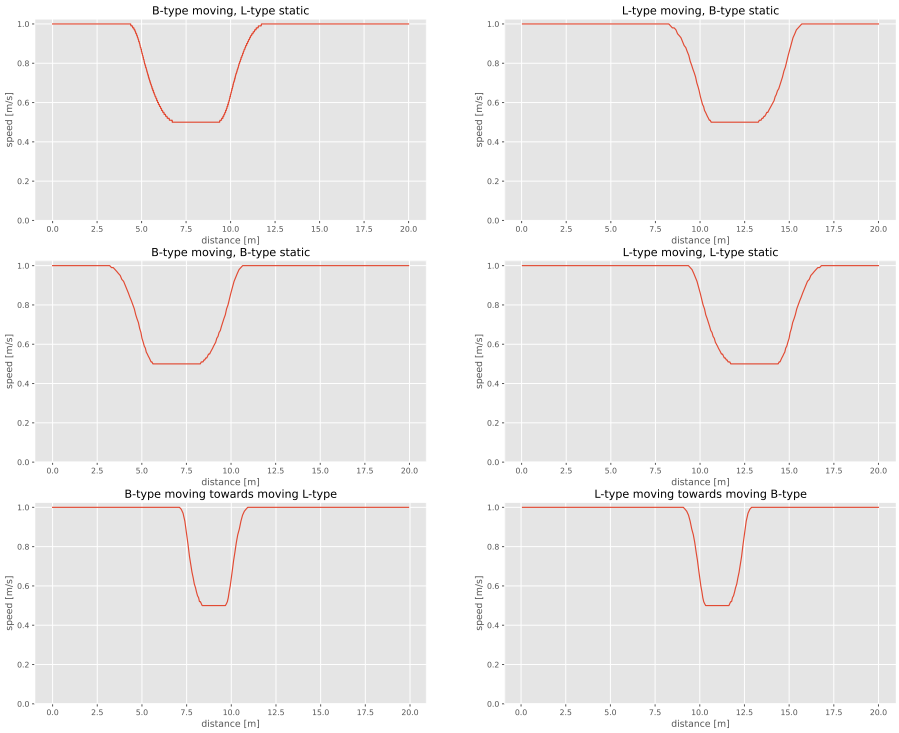

Figure A.3 Velocity profiles for 5 scenarios, triangular kernel deflected backwards used as individual mass of agent, $R_{crit} = 0$

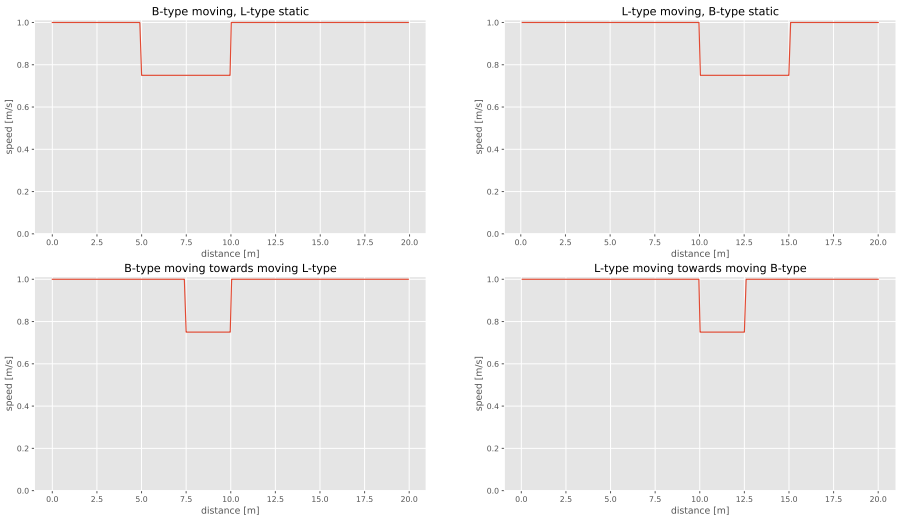

Figure A.4 Velocity profiles for 3 scenarios, triangular kernel deflected forward used as individual mass of agent, $R_{crit} = 0.5$

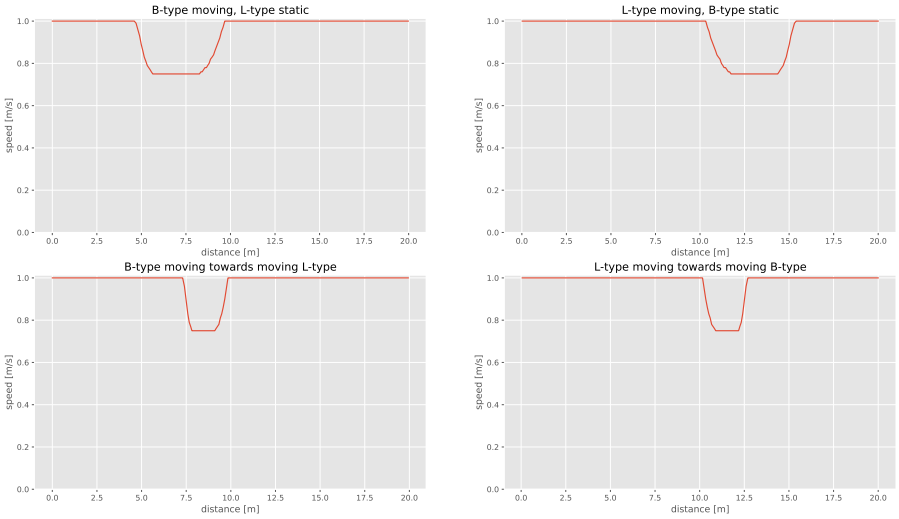

Figure A.5 Velocity profiles for 3 scenarios, triangular kernel deflected backwards used as individual mass of agent, $R_{crit} = 0.5$

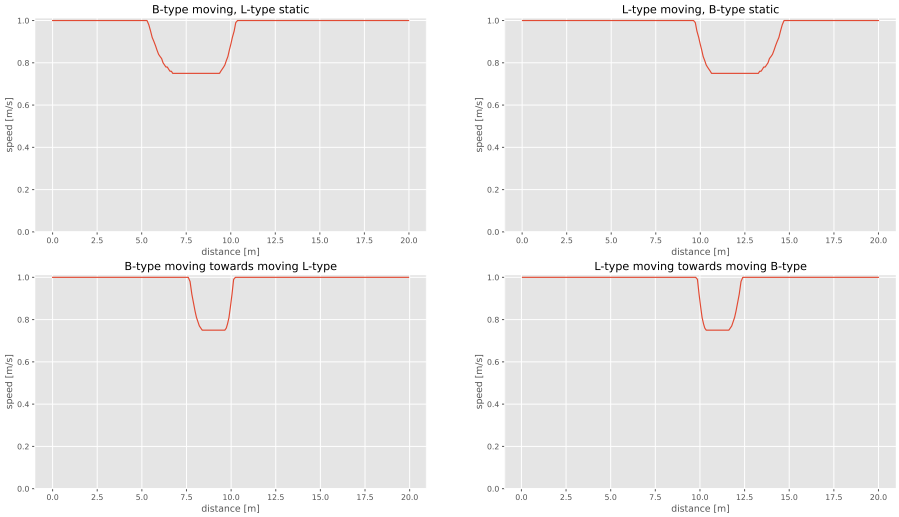

Figure A.6 Velocity profiles for 3 scenarios, triangular kernel deflected backwards used as individual mass of agent, $R_{crit} = 0.5$

A.2 Results of Experiments with Homogeneous Arrival Intensity

This section presents the results for the experiments with the homogeneous arrival intensity that are subject of Section 6.2.2. The experiments analyse the development of two macroscopical characteristics of pedestrian flow, namely average agent count and average agent speed, on the arrival intensity. The experiments were performed for the Diraq kernel, the triangular kernel deflected backward, the triangular kernel deflected backwards, and the kernel that represents free flow.

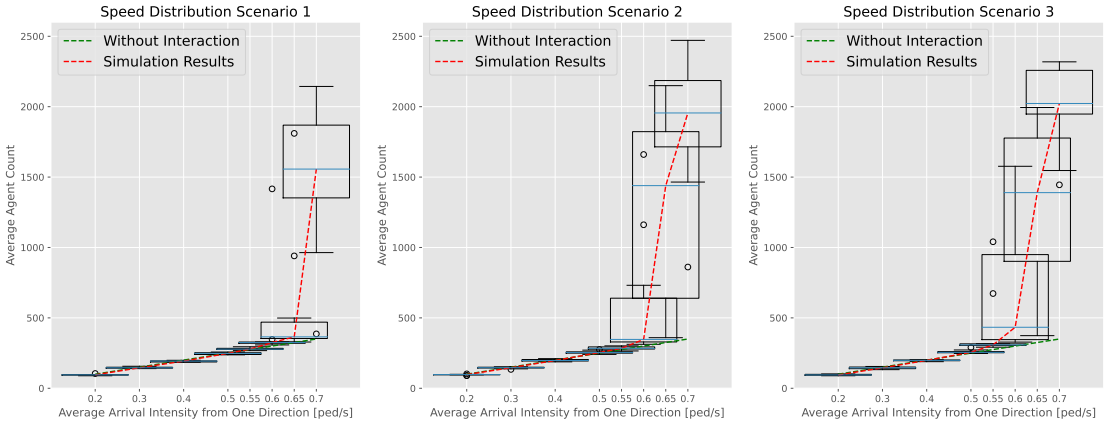

Figure A.7 Development of average agent count in experiments with homogeneous arrival intensity for Diraq kernel

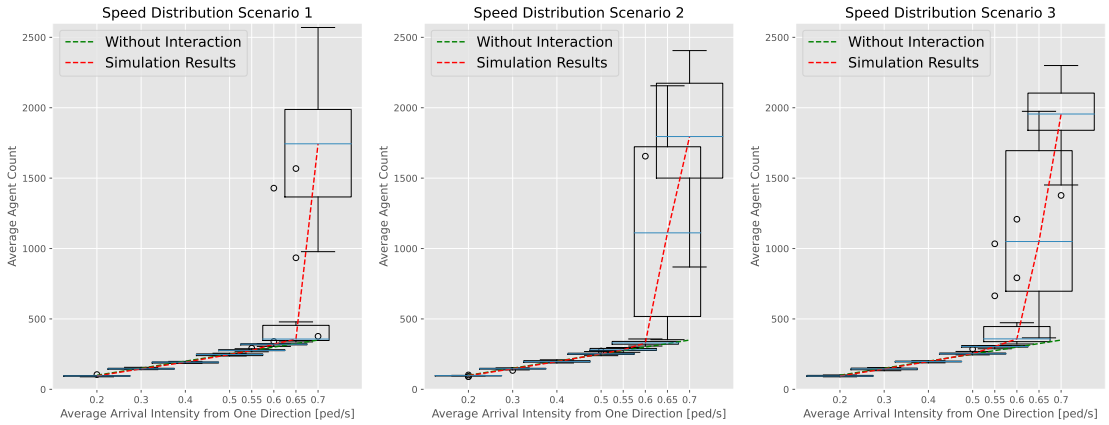

Figure A.8 Development of average agent count in experiments with homogeneous arrival intensity for triangular kernel deflected forward

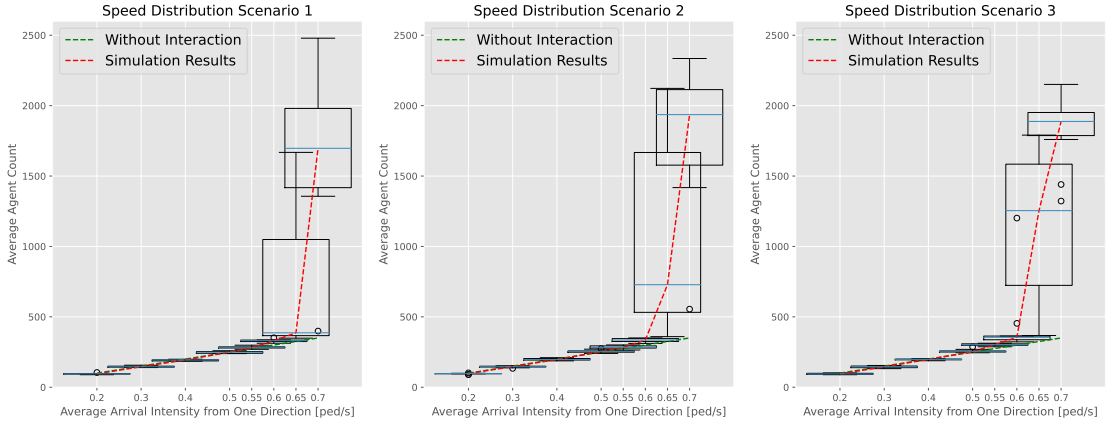

Figure A.9 Development of average agent count in experiments with homogeneous arrival intensity for triangular kernel deflected backwards

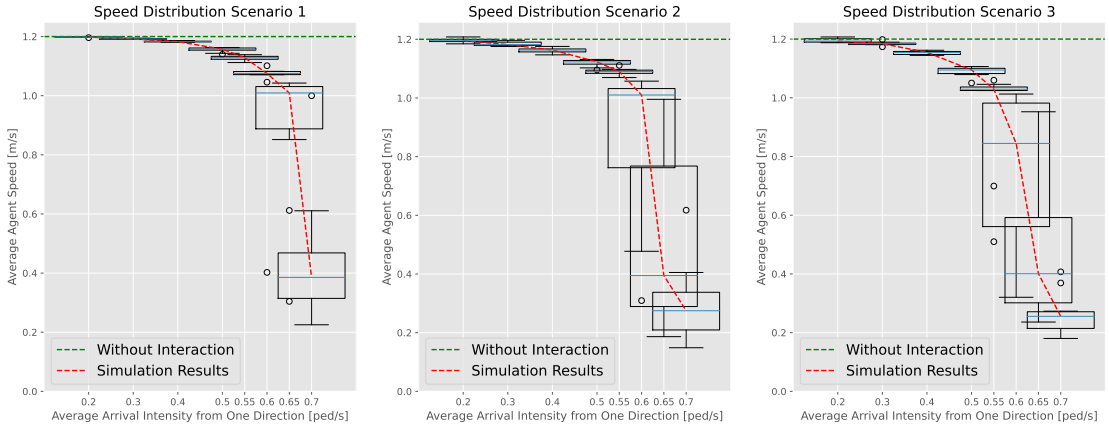

Figure A.10 Development of average speed of agents in experiments with homogeneous arrival intensity for Diraq kernel

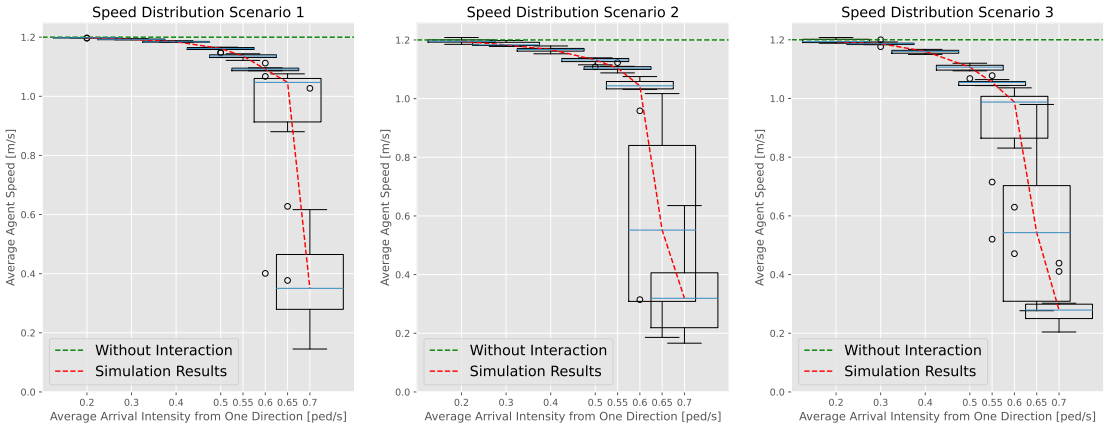

Figure A.11 Development of average speed of agents in experiments with homogeneous arrival intensity for triangular kernel deflected forward

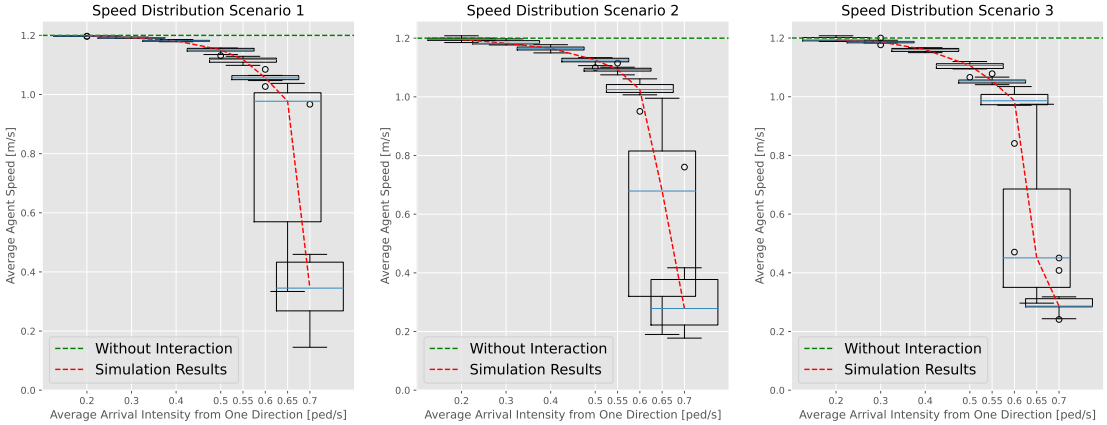

Figure A.12 Development of average speed of agents in experiments with homogeneous arrival intensity for triangular kernel deflected backwards

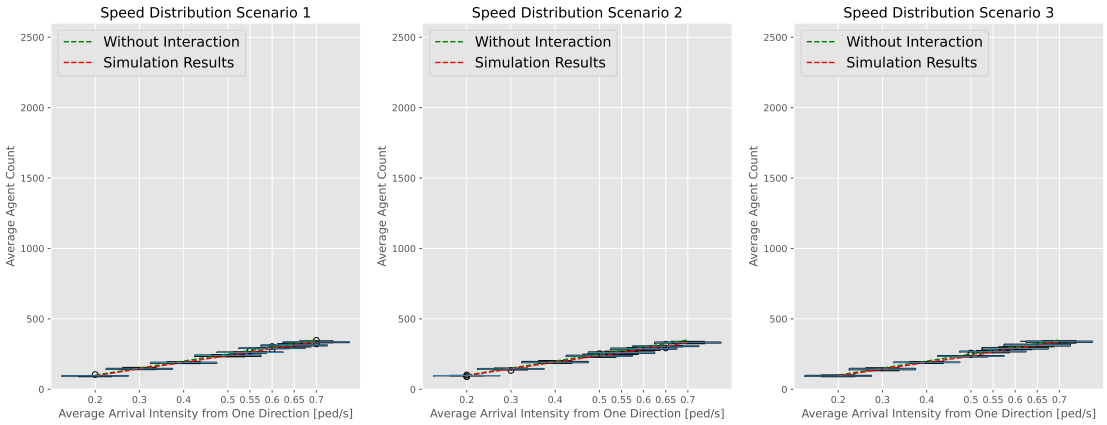

Figure A.13 Development of average agent count in experiments with homogeneous arrival intensity for free flow

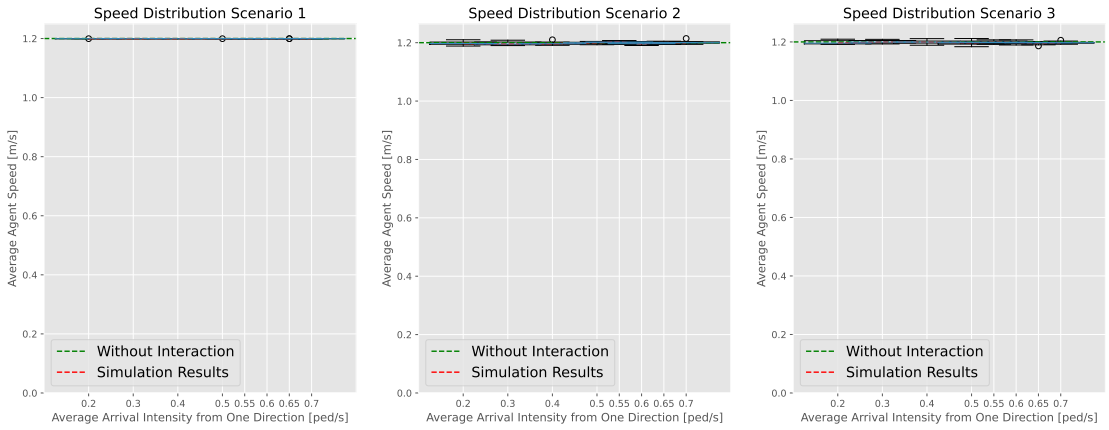

Figure A.14 Development of average speed of agents in experiments with homogeneous arrival intensity for free flow

A.3 Results of Experiments with Non-homogeneous Arrival Intensity

This section presents the results for the experiments with the non-homogeneous arrival intensity that are subject of Section 6.2.3. The experiments analyse the development of two macroscopical characteristics of pedestrian flow, namely average agent count and average agent speed, on the arrival intensity. The experiments were performed for the Diraq kernel, the triangular kernel deflected backward, the triangular kernel deflected backwards, and the kernel that represents free flow.

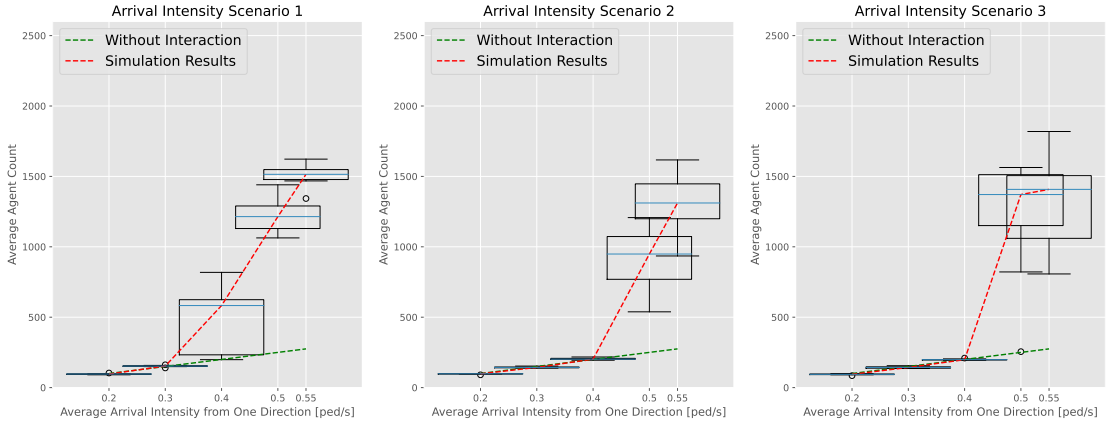

Figure A.15 Development of average agent count in experiments with non-homogeneous arrival intensity for Diraq kernel

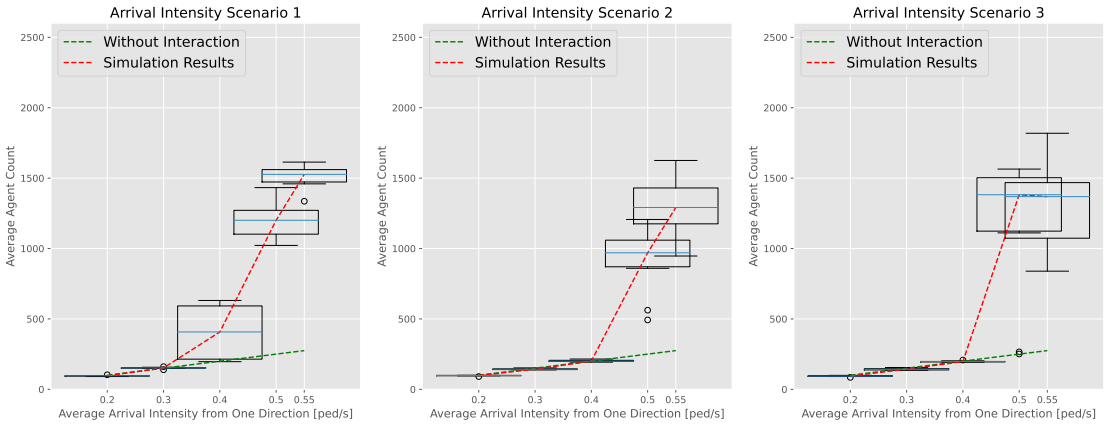

Figure A.16 Development of average agent count in experiments with non-homogeneous arrival intensity for triangular kernel deflected forward

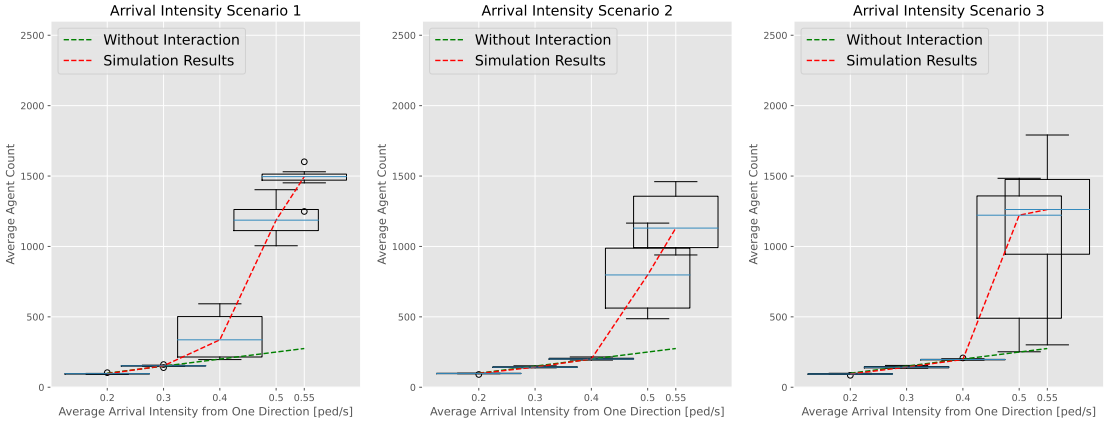

Figure A.17 Development of average agent count in experiments with non-homogeneous arrival intensity for triangular kernel deflected backwards

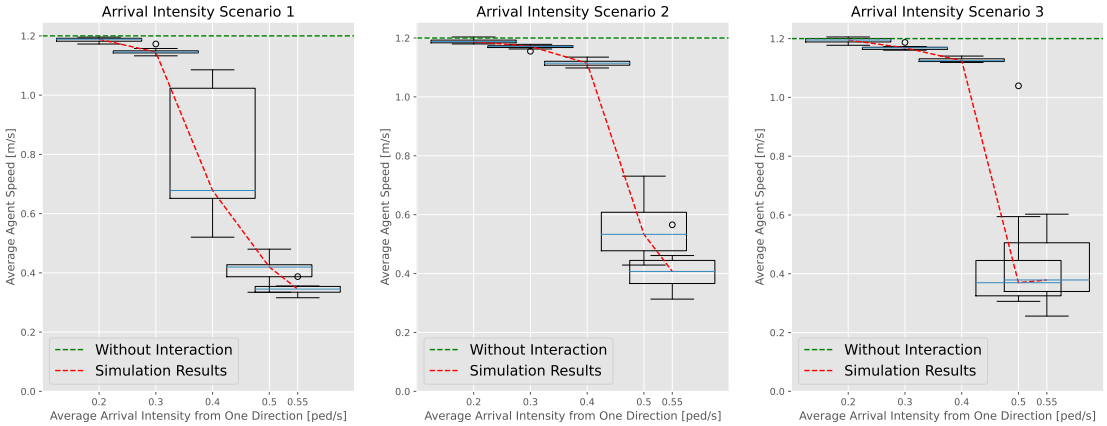

Figure A.18 Development of average speed of agents in experiments with non-homogeneous arrival intensity for Diraq kernel

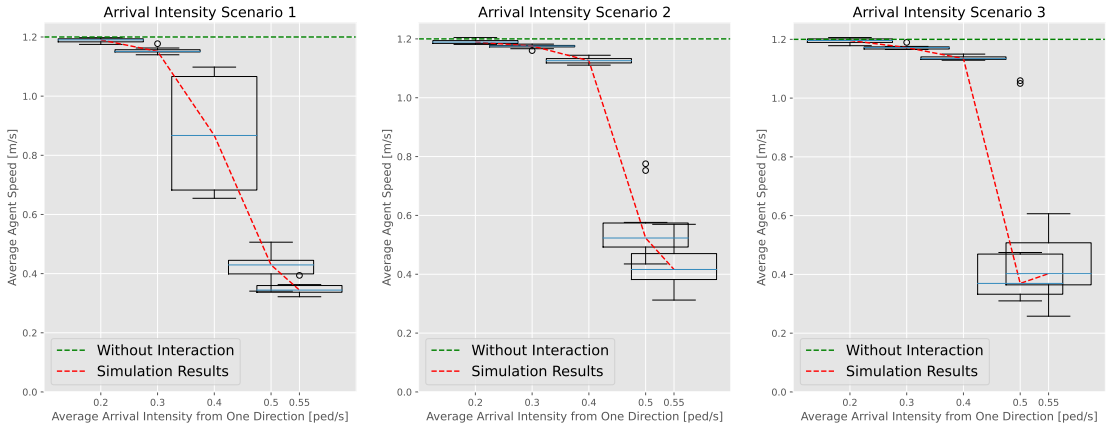

Figure A.19 Development of average speed of agents in experiments with non-homogeneous arrival intensity for triangular kernel deflected forward

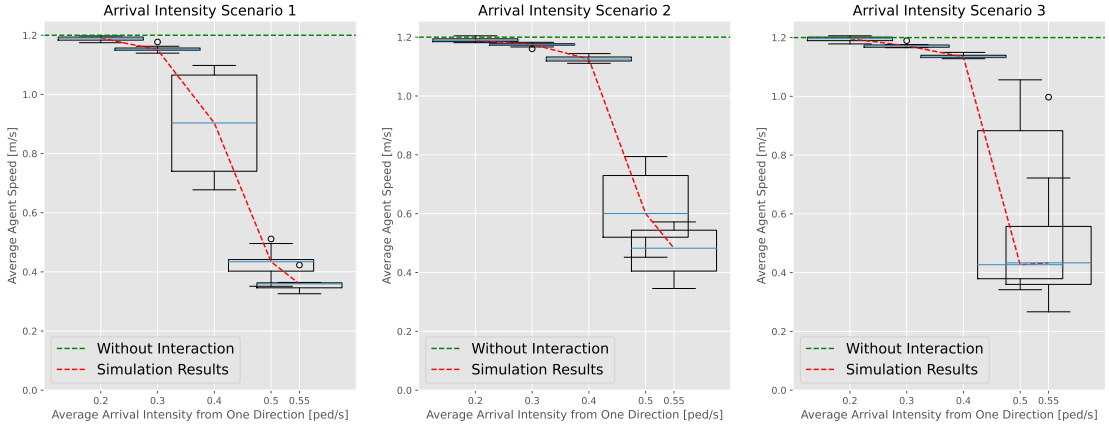

Figure A.20 Development of average speed of agents in experiments with non-homogeneous arrival intensity for triangular kernel deflected backwards

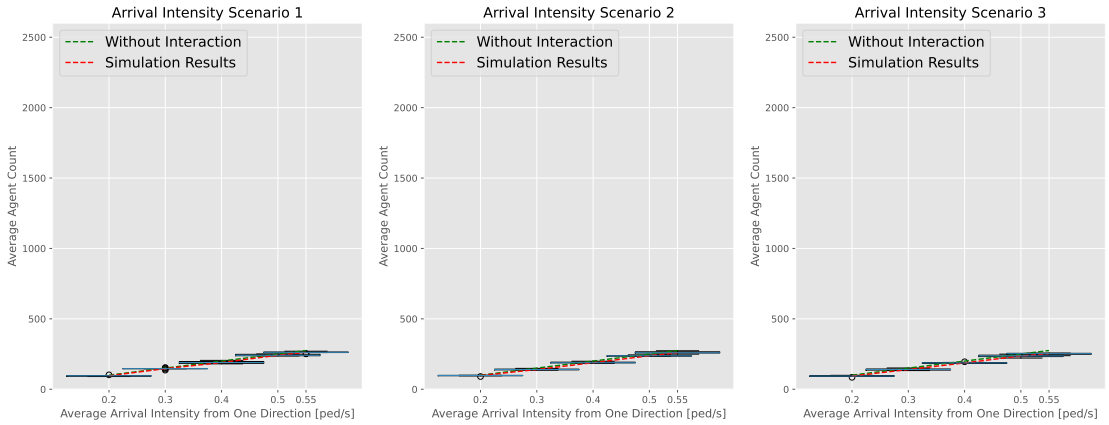

Figure A.21 Development of average agent count in experiments with non-homogeneous arrival intensity for free flow

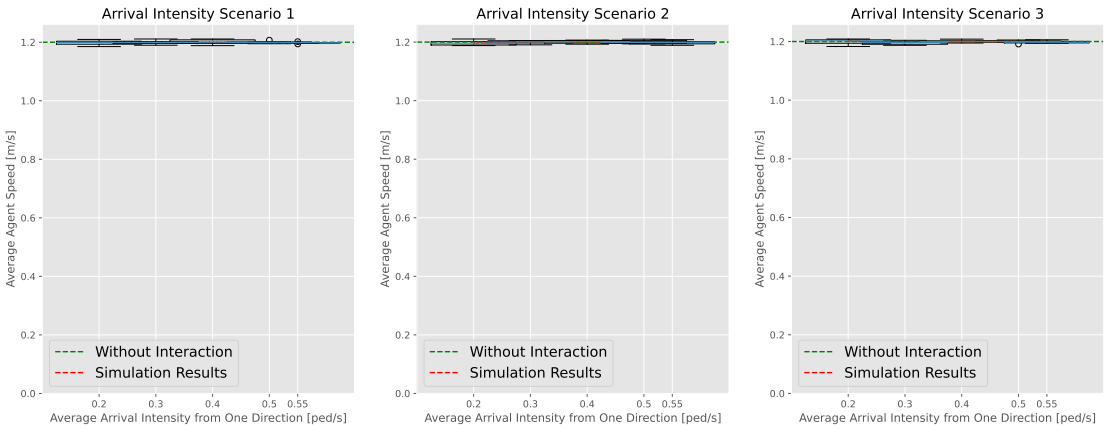

Figure A.22 Development of average speed of agents in experiments with non-homogeneous arrival intensity for free flow

Bibliography

1. VACKOVÁ, Jana; BUKÁČEK, Marek. Kernel estimates as general concept for the measuring of pedestrian density. *Transportmetrica A Transport Science*. 2023. ISSN 2324-9935. Available from DOI: 10.1080/23249935.2023.2236236.
2. RUSSELL, Stuart; NORVIG, Peter. *Artificial Intelligence: A Modern Approach*. 3rd. Upper Saddle River, NJ: Prentice Hall, 2010. ISBN 0-13-461099-7.
3. WOLFRAM, Stephen. *A New Kind of Science*. Wolfram Media, 2002. ISBN 1579550088. Available also from: https://www.wolframscience.com.
4. CONWAY, John Horton. MATHEMATICAL GAMES: The fantastic combinations of John Conway's new solitaire game "life". *Scientific American*. 1970, vol. 223, no. 4, pp. 120–123.
5. SCHADSCHNEIDER, Andreas; CHOWDHURY, Debashish; NISHINARI, Katsuhiro. *Stochastic Transport in Complex Systems: From Molecules to Vehicles*. 1st ed. Amsterdam: Elsevier Science, 2010. ISBN 978-0-444-52853-7.
6. BURSTEDDE, C; KLAUCK, K; SCHADSCHNEIDER, A; ZITTARTZ, J. Simulation of pedestrian dynamics using a two-dimensional cellular automaton. *Physica A: Statistical Mechanics and its Applications*. 2001, vol. 295, no. 3, pp. 507–525. ISSN 0378-4371. Available from DOI: https://doi.org/10.1016/S0378-4371(01)00141-8.
7. HELBING, Dirk; MOLNÁR, Péter. Social force model for pedestrian dynamics. *Phys. Rev. E*. 1995, vol. 51, pp. 4282–4286. Available from DOI: 10.1103/PhysRevE.51.4282.
8. NEUMANN, T. TASEP related models with traffic light boundary. *The European Physical Journal B*. 2009, vol. 67, no. 1, pp. 133–138. ISSN 1434-6036. Available from DOI: 10.1140/epjb/e2009-00010-2.
9. RAJEWSKY, N.; SANTEN, L.; SCHADSCHNEIDER, A.; SCHRECKENBERG, M. The Asymmetric Exclusion Process: Comparison of Update Procedures. *Journal of Statistical Physics*. 1998, vol. 92, no. 1, pp. 151–194. ISSN 1572-9613. Available from DOI: 10.1023/A:1023047703307.
10. HRABÁK, Pavel. *Nehomogenní Poissonův proces* [Online]. 2023. Available also from: https://courses.fit.cvut.cz/NI-VSM/lectures/files/NI-VSM-Lec-23-Slides.pdf. Lecture notes in course NI-VSM on Faculty of Information Technology, Czech Technical University.
11. HOWELL, K.B. *Ordinary Differential Equations: An Introduction to the Fundamentals*. 1st. CRC Press, 2015. Available also from: https://doi.org/10.1201/9780429466090.

12. WEIDMANN, Ulrich. *Transporttechnik der Fussgänger: Transporttechnische Eigenschaften des Fussgängerverkehrs, Literaturauswertung*. 1992-01. Tech. rep. Institut für Verkehrsplanung, Transporttechnik, Strassen- und Eisenbahnbau (IVT), ETH Zürich. Available from DOI: 10.3929/ethz-a-000687810.

13. KRETZ, Tobias. *An overview of fundamental diagrams of pedestrian dynamics*. 2019-10. Tech. rep. PTV Group. Available from DOI: 10.13140/RG.2.2.30070.96326.

14. DUIVES, Dorine C.; DAAMEN, Winnie; HOOGENDOORN, Serge P. Quantification of the level of crowdedness for pedestrian movements. *Physica A: Statistical Mechanics and its Applications*. 2015, vol. 427, pp. 162–180. ISSN 0378-4371. Available from DOI: 10.1016/j.physa.2014.11.054.

15. FRUIN, John J. Designing for Pedestrians: A Level-of-Service Concept. *Highway Research Record*. 1971, vol. 355, pp. 1–15. ISBN 0309019680. Available also from: http://onlinepubs.trb.org/Onlinepubs/hrr/1971/355/355-001.pdf.

16. EDIE, L. Discussion of Traffic Stream Measurements and Definitions. 1963, pp. 139–154.

17. STEFFEN, B.; SEYFRIED, A. Methods for measuring pedestrian density, flow, speed and direction with minimal scatter. *Physica A: Statistical Mechanics and its Applications*. 2010, vol. 389, no. 9, pp. 1902–1910. ISSN 0378-4371. Available from DOI: 10.1016/j.physa.2009.12.015.

18. PUSHKAREV, B.; ZUPAN, J. M. CAPACITY OF WALKWAYS. *Transportation Research Record*. 1975, vol. 538, pp. 1–15. ISBN 0309023890. ISSN 0361-1981. Available also from: http://onlinepubs.trb.org/Onlinepubs/trr/1975/538/538-001.pdf.

19. NAVIN, F. P. D.; WHEELER, R. J. Pedestrian Flow Characteristics. *Traffic Engineering*. 1969, vol. June, pp. 30–36.

20. ITAMI, Robert M. *Estimating Capacities for Pedestrian Walkways and Viewing Platforms*. 22 Dunstan Avenue, Brunswick, Victoria 3056, Australia, 2002-06. GeoDimensions Pty Ltd. Available also from: https://cales.arizona.edu/~gimblett/Estimating%20Pedestrian%20Capacities%20for%20Walkways.pdf.

21. HRABÁK, Pavel. Time-headway distribution for random-sequential-update TASEP with periodic and open boundaries. *Journal of Traffic and Transportation Engineering (English Edition)*. 2020, vol. 7, no. 1, pp. 30–41. ISSN 2095-7564. Available from DOI: 10.1016/j.jtte.2019.03.006. Special Issue: Modeling and detecting traffic dynamics: granular, pedestrian and vehicular flow.

22. HUFF, K. Herbie; LIGGETT, Robin. *The Highway Capacity Manual's Method for Calculating Bicycle and Pedestrian Levels of Service: the Ultimate White Paper*. 3250 Public Affairs Building, Los Angeles, CA, United States, 90095-1656, 2014-09. Tech. rep. Lewis Center for Regional Policy Studies, University of California, Los Angeles, Institute of Transportation Studies. ISBN 0309023890. Available also from: http://www.lewis.ucla.edu/wp-content/uploads/sites/2/2014/09/HCM-BICYCLE-AND-PEDESTRIAN-LEVEL-OF-SERVICE-THE-ULTIMATE-WHITE-PAPER.pdf.

23. SAHANI, Rima; BHUYAN, P.K. Modelling Pedestrian Perspectives in Evaluating Satisfaction Levels of Urban Roadway Walking Facilities. *Transportation Research Procedia*. 2020, vol. 48, pp. 2262–2279. ISSN 2352-1465. Available from DOI: 10.1016/j.trpro.2020.08.289. Recent Advances and Emerging Issues in Transport Research – An Editorial Note for the Selected Proceedings of WCTR 2019 Mumbai.

24. MANNING, Robert E. *Studies in Outdoor Recreation: Search and Research for Satisfaction*. Fourth Edition, Fourth. Oregon State University Press, 2022. ISBN 0870712098.

25. LAWSON, Steven R. Computer Simulation as a Tool for Planning and Management of Visitor Use in Protected Natural Areas. *Journal of Sustainable Tourism*. 2006, vol. 14, no. 6, pp. 600–617. Available from DOI: 10.2167/jost625.0.

26. STEVEN R. LAWSON Robert M. Itami, H. Randy Gimblett; MANNING, Robert E. Benefits and Challenges of Computer Simulation Modeling of Backcountry Recreation Use in the Desolation Lake Area of the John Muir Wilderness. *Journal of Leisure Research*. 2006, vol. 38, no. 2, pp. 187–207. Available from DOI: 10.1080/00222216.2006.11950075.

27. COLE, David N. *Computer simulation modeling of recreation use: Current status, case studies, and future directions*. 2005. Available from DOI: 10.2737/rmrs-gtr-143.

28. TORRENS, Paul M.; NARA, Atsushi; LI, Xun; ZHU, Haojie; GRIFFIN, William A.; BROWN, Scott B. An extensible simulation environment and movement metrics for testing walking behavior in agent-based models. *Computers, Environment and Urban Systems*. 2012, vol. 36, no. 1, pp. 1–17. ISSN 0198-9715. Available from DOI: https://doi.org/10.1016/j.compenvurbsys.2011.07.005.

29. CAMPANELLA, Mario. *Microscopic Modelling of Walking Behaviour*. Delft, the Netherlands, 2016. ISBN 978-90-5584-214-8. Available from DOI: 10.4233/uuid:b65e6e12-a85e-4846-9122-0bb9be47a762. Dissertation. TRAIL Research School. TRAIL Thesis Series T2016/20.

30. ZHANG, Jun; SEYFRIED, Armin. Quantification of Bottleneck Effects for Different Types of Facilities. *Transportation Research Procedia*. 2014, vol. 2, pp. 51–59. ISSN 2352-1465. Available from DOI: 10.1016/j.trpro.2014.09.008. The Conference on Pedestrian and Evacuation Dynamics 2014 (PED 2014), 22-24 October 2014, Delft, The Netherlands.

31. LINDSEY, Patrick; LINDSEY, Greg. Using Pedestrian Count Models to Estimate Urban Trail Traffic. *The Journal of Regional Analysis and Policy*. 2004, vol. 34, pp. 50–68. Available also from: https://www.researchgate.net/publication/265669039_Using_Pedestrian_Count_Models_to_Estimate_Urban_Trail_Traffic.

32. WANG, Jueyu; HANKEY, Steve; WU, Xinyi; LINDSEY, Greg. Monitoring and Modeling of Urban Trail Traffic: Validation of Direct Demand Models in Minneapolis, Minnesota, and Columbus, Ohio. *Transportation Research Record*. 2016, vol. 2593, no. 1, pp. 47–59. Available from DOI: 10.3141/2593-06.

33. ERMAGUN, Alireza; LINDSEY, Greg; HADDEN LOH, Tracy. Bicycle, pedestrian, and mixed-mode trail traffic: A performance assessment of demand models. *Landscape and Urban Planning*. 2018, vol. 177, pp. 92–102. ISSN 0169-2046. Available from DOI: 10.1016/j.landurbplan.2018.05.006.

34. ZHANG, J.; KLINGSCH, W.; SCHADSCHNEIDER, A.; SEYFRIED, A. Ordering in bidirectional pedestrian flows and its influence on the fundamental diagram. *Journal of Statistical Mechanics: Theory and Experiment*. 2012, vol. 2012, no. 02, P02002. Available from DOI: 10.1088/1742-5468/2012/02/P02002.

35. ŠTEKEROVÁ, Kamila; ZELENKA, Josef; KOŘÍNEK, Milan. Agent-Based Modelling in Visitor Management of Protected Areas. *Sustainability*. 2022, vol. 14, no. 19. ISSN 2071-1050. Available from DOI: 10.3390/su141912490.

36. GINTERS, Egils; SILINS, Artis. Multi-level approach for environmental systems modelling in the Ligatne Natural Trails. *Sociotechnical Systems Engineering Institute*. 2007, vol. 6, pp. 795–801. Available also from: https://www.researchgate.net/publication/297118169_Multi-level_approach_for_environmental_systems_modelling_in_the_Ligatne_Natural_Trails.

37. RONGXU QIU, Wei Xu; LI, Shan. Agent-based Modeling of the Spatial Diffusion of Tourist Flow—A Case Study of Sichuan, China. *Journal of China Tourism Research*. 2016, vol. 12, no. 1, pp. 85–107. Available from DOI: 10.1080/19388160.2016.1160847.

38. GIMBLETT, H.R.; RICHARDS, M.T.; ITAMI, R.M. RBSim: Geographic Simulation of Wilderness Recreation Behavior. *Journal of Forestry*. 2001, vol. 99, no. 4, pp. 36–42. ISSN 0022-1201. Available from DOI: 10.1093/jof/99.4.36.

39. ZHANG, Sainan; ZHANG, Jun; CHRAIBI, Mohcine; SONG, Weiguo. A speed-based model for crowd simulation considering walking preferences. *Communications in Nonlinear Science and Numerical Simulation*. 2021, vol. 95, p. 105624. ISSN 1007-5704. Available from DOI: 10.1016/j.cnsns.2020.105624.

40. SAEED, R.A.; RECUPERO, Diego Reforgiato; REMAGNINO, Paolo. Simulating crowd behaviour combining both microscopic and macroscopic rules. *Information Sciences*. 2022, vol. 583, pp. 137–158. ISSN 0020-0255. Available from DOI: 10.1016/j.ins.2021.11.028.

41. SUN, Yi. Kinetic Monte Carlo simulations of two-dimensional pedestrian flow models. *Physica A: Statistical Mechanics and its Applications*. 2018, vol. 505, pp. 836–847. ISSN 0378-4371. Available from DOI: https://doi.org/10.1016/j.physa.2018.04.017.

42. CUI, Geng; YANAGISAWA, Daichi; NISHINARI, Katsuhiro. Learning from experimental data to simulate pedestrian dynamics. *Physica A: Statistical Mechanics and its Applications*. 2023, vol. 623, p. 128837. ISSN 0378-4371. Available from DOI: https://doi.org/10.1016/j.physa.2023.128837.