Computational decision support for crowd management applications

A case study on operational in-event pedestrian crowd management

Master thesis submitted to Delft University of Technology in partial fulfilment of the requirements for the degree of **MASTER OF SCIENCE** in **Engineering and Policy Analysis**

Faculty of Technology, Policy and Management

by

F.T. Boendermaker

Student number: 4655605

To be defended in public on July 7th 2022

Graduation committee

Chairperson & first supervisor:	Dr. C.N. van der Wal,	TU Delft,	Section Systems Engineering
Second supervisor:	Dr.ir. W.L. Auping,	TU Delft,	Section Policy Analysis
External supervisor:	Dr.ir. D.C. Duives,	TU Delft,	Civil Engineering and Geosciences
External supervisor:	Dr. C. Kuster,	Geodan,	

Associated code to this thesis and constructed models are made publicy available at
https://github.com/floristevito/CrowdSim

TU Delft

Preface

This thesis research proposes, and applies, a methodology to provide support for in-event crowd management. This work is written to conclude my degree in Engineering and Policy Analysis at the TU Delft, and performed in collaboration with Geodan.

I started my research in January, in close consult with my first supervisor, Natalie van der Wal. Together, we had long and interesting discussions on crowd management and the application of agent-based models in this field. Eventually, with the extensive input of Willem Auping and Dorine Duives, my second and external supervisor, we shaped the final direction of this research. I want to thank all three supervisors for their reoccurring help, not only by providing elaborate and constructive feedback, but also by being trusted experts to discuss new ideas with.

My other gratefulness goes to Geodan, who has been a constant partner in my work. Most especially Corentin Kuster, my other external supervisor, who provided me with a large amount of useful feedback and ideas to bring this research further. Additionally, I want to send my regards to the entire research department of Geodan, many of whom have read my work along the way and were always available for help.

In addition, I want to thank the research group of Gerta Köster, who are behind the development of the simulation framework Vadere, from the department of Computer Science and Mathematics at the Munich University of Applied Sciences. Their help on using Vadere, and interest in the EMA Workbench, has given me the additional help and motivation to bring this research to what it is right now. Most especially, I want to thank Christina Mayr, who has been of incredible help on the usage of Vadere, and with whom I have spent many occasions with on debugging Vadere specifics.

Finally, I want to thank my friends and family. With special regard to my colleague-student Bruno Hermans, for being my trusted deliberation partner when ideas or techniques had to be considered that were challenging to me. Bruno and I have worked together during both my Bachelor and Master studies in Delft on a constant basis, and our trusted principle has always been: one set of eyes is never enough. Lastly, I want to thank Amber Hessels, who has been my personal editor along every typed word.

F.T. Boendermaker
Delft, 2022

Executive summary

In-event crowd management is a crucial element for hosts of large-scale events and local decision makers to keep situations, such as large concerts or busy shopping streets, safe. Operational support assisting crowd management in practice, however, is sparse. While crowd modelling has the potential to solve this lack of information by illustrating effects of measures before they have to be implemented in real life, few model-based studies currently assist operational crowd management applications. This work points to two reasons for the lack of current support: (1) modelling and simulating realistic human crowd behavior is complex; and (2) computational requirements of crowd models are high.

This research presents three novel methodological steps for experimentation—exploration, selection, and evaluation—to deal with the complexity and uncertainties inherent to crowd modelling and the computational requirements. By (1) constructing a microscopic crowd model using the Vadere simulator, and (2) using techniques from the field of exploratory modelling and analysis, this work showcases how useful insights for crowd managers can be generated based on the proposed methodology. A connector between Vadere and the Python based Exploratory Modelling and Analysis (EMA) Workbench was constructed to synergize these two steps. The proposed methodology is then applied to the case of the shopping street 'Grote Markt', in the city of Breda, The Netherlands. This work aims to answer the question: *"what are the potential effects of three proposed in-event crowd management measures—traffic regulators, directional guidance, and object placement—on the pedestrian speed and densities?"*.

Findings highlight the potential of the proposed traffic regulator measure in the case study of the Grote Markt, and its effectiveness compared to object placement and directional guidance. The traffic regulators measure successfully increased the pedestrian speed and lowered the density in problematic areas, often preventing the safety threshold of more than one pedestrian per square meter from being reached. The applied methodological steps for experimentation furthermore lead to a better understanding of in-event crowd management at the Grote Markt, including a wide range of possible favorable and unfavorable circumstances. For instance, problematic situations are to be expected when pedestrians strive for a large amount of personal space in combination with a high influx of visitors from the Veemarktstraat or the south side of the Grote Markt. The applied methodology furthermore provided a way to deal with the many uncertainties in the field of crowd simulations, and managed to mind computational limitations by presenting a way to quickly search through a wide range of circumstances.

This research concludes that there is high potential for the proposed methodological steps to aid operational support on crowd management in complex case studies such as the investigated shopping street in Breda. The proposed steps enabled the utilization of complex crowd models to better understand in-event crowd management systems, and thereby provided the comparison between different in-event measures before they have to be implemented in real life. Limitations of this work include the reliance on underlying model specific assumptions, and practicalities regarding the methodological application for crowd managers. However, the most significant theoretical implication of this work is the foundation of a methodology to provide operational support on crowd management applications, and to use microscopic pedestrian crowd models in the process. Most of all, with additional practical guidance, the methodology of this work can be applied to more operational in-event crowd management cases, and significantly help crowd managers in various sectors.

List of Tables

List of Figures

Contents

Glossary

box Specifies a specific location in the uncertainty space. Illustrated by restricted dimensions, belonging together, giving a range of values for each restricted dimension. 46

case A combination of model input parameters. 8, 20, 41, 45, 55, 67

epsilon value Specifies the granularity of the grid used to find solution in multi-objective optimization. Higher values generally lead to a lower set of solutions maintained in the Pareto set. 43

NFE The number of evaluations used in optimization. This number can be increased to obtain stabilized results. 43

raw Referring to an unmodified model outcome value. In this study, outcome values are often preprocessed, for example to calculate the robustness values. Raw values thus denote the outcomes before processing. Note that the averaging over a number of replications is however performed before inspecting this raw outcome value, so the values are not truly and completely raw. 46

run An evaluation of a model case on defined outcomes. Optionally post-processed over multiple replications. 6, 18, 22, 40, 45, 59, 67

runtime The duration of performing a model run. 22, 42, 50, 67

scenario A specific location in the uncurtainty space that leads to a certain group of model outcomes. 2, 39, 45, 55

scenario file Referring to a Vadere (simulation framework) specific scenario file. A Vadere scenario includes all model components, including the model structure and runtime settings. 24, 67

seed Refers to the random seed that initiates the random generator. With a set seed, the same results can be acquired from a (pseudo) random function. 20, 67

zero case A reference case where no crowd management measure is applied. Used to compare performance of measures against a situation where no crowd management would be aplied under the same circumstances. 43, 49, 57

1 Introduction

Highlighted by the tragic event of the crowd crush at the Astroworld festival (The Guardian, 2021), crowd management is a crucial element for hosts of large-scale events to keep situations safe (Higgins, n.d.; Speakman, 2021). Outside typical large-scale events, crowd management is just as crucial for local decision makers, for instance to keep shopping streets safe with the influxes of visitors. Especially after the recent relaxation in COVID-19 related restrictions (NOS, 2021). When shopping streets get too crowded, potentially unsafe situations could occur, such as the dreadful turn of events in Almere Stad, where two men were abused in a shopping area after bumping into one another pedestrian by accident (Omroep Flevoland, 2022). Hosts of events and local decision makers need operational support to assist with crowd management in practice, but applications of this kind are sparse (Wijermans et al., 2016).

Crowd modelling and simulation has the potential to aid decision-making in crowd management (Sharma et al., 2018). Simulations enable the understanding of crowd management systems, showing insights into crowd management measure effects before they have to be implemented in practice. Often, great attention is given to the preparation of a certain event, while it remains clear that preparation alone is not sufficient and support during events remains crucial (Wijermans et al., 2016). In particular when preparation is more difficult, for instance due to uncertainty about the number of pedestrians or their behavior beforehand, in-event crowd management is crucial. This is the phase where hosts or local decision-makers can intervene and implement measures when the situation gets out of hand. Crowd simulation can significantly aid here by helping to understand the in-event crowd management systems, thereby generating insights into what in-event crowd management measures show potential for real life application.

This work points to two potential reasons for the lack of in-event crowd management insights: (1) modelling and simulating realistic human crowd behavior is complex (Pelechano et al., 2007), and bound to cultural influences (Kaminka & Fridman, 2018), inevitably leading to various field specific applications (Koukopoulos et al., 2018; Shende et al., 2011; Teo et al., 2015); and (2) computational requirements of crowd models are often high, or lack microscopic behavioral elements (Duives et al., 2013). In order to fully utilize crowd models for operational insights into in-event crowd management, both challenges have to be addressed. This research will therefore propose a novel set of methodological steps regarding model-based experimentation for in-event crowd management.

To propose this set of functional methodological steps, needed to bridge the gap between building crowd models and utilizing them for operational insights, efforts from multiple fields need to be synergized. Crowd management is a grand societal challenge, a global problem that can potentially be solved by collective, inter-field, collaboration (George et al., 2016). On the one hand, extensive and sophisticated crowd models, developed by field experts, are needed in order to simulate pedestrian crowds realistically. On the other hand, efforts from other fields are needed to help utilize these models, putting them to practice and helping crowd management stakeholders in understanding their crowd management systems more thoroughly. In particular when focusing on inner-city crowds, effective management is crucial in order to preserve a sustainable area that is safe and accessible for all pedestrians.

This research therefore aims to apply the proposed methodology on a case study of practical crowd management in an inner-city area. This will be done using the case of the shopping street 'Grote Markt' during a regular busy weekend shopping day, in the city of Breda, The Netherlands. This area is known for its high density of bars and terraces ("7x zomerse terrassen in Breda!", 2018), limiting walkable space for pedestrians. In this case study, pre-event crowd management preparation is difficult, since there is not necessarily a particular event happening, bounded for instance by a certain number of visitors. Therefore, this research focuses on generating insights into the in-event crowd management system, referring to crowd management during the shopping day itself. The notion of in-event here refers not only to planned and specific types of events, but also refers to typical, busy, shopping occasions.

A well-suited approach to propose and apply the methodological steps is twofold: (1) building a detailed, agent-based, microsporic pedestrian crowd model; and (2) using tools from exploratory modelling and analyses (Bankes, 1993) to utilize this model for better understanding of the in-event crowd management system, and thereby generating operational insights for crowd managers. Performing both steps allows for the synergizing between the fields of the crowd modelling experts and operational crowd managers. Thereby bridging the gap between building crowd models, and utilizing them for operational decision-support.

Regarding the first part, model building, an agent-based modelling (ABM) method, as seen in similar works (Haciomeroglu et al., 2008; Teo et al., 2015) and discussed in detail by Wilensky and Rand (2015), is well-suited. The utilization of an ABM approach allows for a valid integration of the complex behavior, and individual decision-making, of pedestrians. Especially due to the ability of ABMs to model individuals heterogeneously (Wilensky & Rand, 2015, p. 32). This element is crucial, for instance, when pedestrians in the simulation model form groups and change their behavior accordingly.

Because a sophisticated model is needed for the application of methodological steps in a relative limited timeframe, this research makes use of two practical solutions. First, this work will be in collaboration with field experts from Geodan, who are involved in the EU funded smart city monitoring project based in Breda (Argaleo, 2021). Geodan will help this research by providing field specific insights and guidance. Second, simulation framework Vadere (Kleinmeier et al., 2019) will be used for model construction. Vadere provides extensive out-of-the-box support of many pedestrian crowd dynamic mechanisms, such as different locomotion models and a group model, and enables this research to construct a microscopic crowd model of the Grote Markt rapidly.

For the second methodological part, the utilization of exploratory modelling and analysis is needed to allow for the incorporation of the large uncertainty space surrounded by crowd modelling, while keeping computational efforts feasible. In order to apply techniques from the field of exploratory modelling to crowd management, this work will use the Exploratory Modelling and Analysis (EMA) Workbench (Kwakkel, 2017). This is a workbench implemented in Python that enables easy use of several extensive exploratory modelling tools, like scenario discovery and sensitivity analysis. Because the Vadere simulator has never been used in combination with the EMA Workbench before, this research will build a new connector between the two.

This research aims to not only demonstrate how a crowd model can be built, but most especially how such a model can be utilized for operational insights regarding in-event crowd management. By the application of the proposed methodological steps for experimentation, this research will analyze the effect of in-event crowd management on both the speed and density of pedestrians on the Grote Markt. Three crowd management measures, based on related works (Karbovskii et al., 2019; Park et al., 2018; Pellegrini et al., 2009), are applied specifically to the case study: (a) traffic regulators that can be deployed in the area of interest and instruct pedestrians to move differently; (b) directional guidance, which represents a communication method, for example signs, instructing pedestrians to take a certain route or change their destination; and (c) object placement, which places line barriers in the area of interest. With this knowledge, the following research questions are proposed:

What effect do the in-event crowd management measures—traffic regulators, directional guidance, and object placement—have on the density and walking speed of pedestrians in the Grote Markt, Breda?

Sub-question 1. *What are the roles and limitations of crowd models for in-event crowd management?* In the first stage, a literature study, that will be presented in the first half of Chapter 2, will give an insight into the place of crowd models in the current in-event crowd management landscape, together with their limitations. This will help to get an understanding on how crowd models can be utilized optimally. The answer to this question will lead to a high-level overview of the methodological steps needed for model-based experimentation, with the aim to support operational crowd management.

Sub-question 2. *How can in-event crowd management at the Grote Markt be modelled?*
This question will be covered by applying relevant literature on the case study of the Grote Markt in the second half of Chapter 2, and by the model construction that will be presented in Chapter 3. The answer to this question will subsequently cover how in-event crowd management measures and the quantification of their performance at the Grote Markt can be implemented in a crowd model.

Sub-question 3. *How can a microscopic crowd model of the Grote Markt, that includes group forming and realistic pedestrian movements, aid operational crowd management?*
This question is targeted at the model-based experimental setup. The answer to this question will be discussed in Chapter 4, and represent a detailed overview of proposed methodological steps for model experimentation for the Grote Markt case study.

Sub-question 4. *How do the crowd management measures at the Grote Markt perform, given the inherent uncertainty of crowd modelling?*
Finally, this research will conclude with the application of the proposed methodological steps for model experimentation. Results will be presented in Chapter 5, and discussed and concluded in Chapter 6. Results will give an insight into the system of in-event crowd management at the Grote Markt, and illustrate the effects of the three proposed crowd management measures.

The next part of this work, Chapter 2, includes an extensive theoretical background into in-event crowd management. The constructed model for the case study is presented in Chapter 3, and a detailed overview of the proposed methodological steps for model-based experimentation is given in Chapter 4. Results of application of the methodology on the case study are shown in Chapter 5, and discussed and concluded in Chapter 6.

2

Theoretical background

This chapter presents a theoretical background in the area of in-event crowd management and the operational support thereof. A clear distinction is made between the overarching generic literature components and the components of literature that are specifically applied to the case study of interest within this research: the Grote Markt. The first section discusses the roles and limitations of crowd models for in-event crowd management, and is thereby targeted at answering the first research sub question. Afterwards, literature component are applied specifically to the case study of the Grote Markt, providing a foundation for the model construction, presented in Chapter 3.

2.1. Crowd models for in-event crowd management

This section discusses elements that are involved in, or with, crowd models for in-event crowd management, which can be translated to crowd management during situations take place. The notion of in-event here refers not only to planned and specific types of events, but can also refer to a busy day in a shopping area. First, the place and need for crowd models for operational support on current crowd management applications are discussed. Afterwards, the complexity of crowd modelling is illustrated by decomposing a crowd model into two important underlying model components: case study specific influences on crowd behavior and the modelling of pedestrians movement. This section concludes with remaining challenges of using crowd models for operational in-event crowd management support.

2.1.1. The need for crowd models and their computational challenges

Extensive and innovative crowd management platforms are growing in popularity, with the aim of optimizing resources and improving citizens lives (Garcia-Retuerta et al., 2021). Crowd counting and tracking methods are extensively covered in literature (Asimakopoulou & Bessis, 2011; Sharma et al., 2018; Suganya et al., 2019). Moreover, recent research proposes solutions for detailed 3D visualizations of crowd management (Yu et al., 2021). Even more so, Haciomeroglu et al. (2008) show that it would be possible to generate a large portion of urban flow data, even in situations where sensory data gathering is limited.

However, challenges in this context are often not with data-collecting methods itself, but with the analysis that must follow (Garcia-Retuerta et al., 2021). This problem can be observed domain wide, as Tao (2013) likewise describes the challenge of Geographical Information Systems (GIS) for smart cities to deal with new data sets of urban movement more intelligently. Overall, recent advances in data gather methods for crowds make it possible to generate vast amounts of data concerning residents in a city, but the challenge often comes down to the processing and analysis of the data (Kaiser et al., 2018).

Recent findings advocate the need for more and better ways to act on the growing collections of urban data, especially with regard to in-event crowd management. Crowd modelling is a tool that could help in this regard (Sharma et al., 2018). By using data and knowledge on pedestrians crowds for realistic models, simulations of these models could enable better understanding of crowd management systems and help determine the potential of crowd management methods. These crowd models are improving, helped for instances by advances in predicting algorithms for crowds and their future states

(Zhang et al., 2018). Detailed models and simulations thereby enable the understanding of crowd management systems, showing insights into in-event crowd management effects before they have to be implemented in practice.

One apparent current problem with the utilization of crowd models is, however, related to computational effort. Often, crowd models are either significantly demanding to run, or models lack microscopic behavior elements (Duives et al., 2013). Most current research on crowd modelling is often focused on field-specific applications such as evacuations (Shende et al., 2011; Teo et al., 2015), police interventions (Park et al., 2018), or carnival parades (Koukopoulos et al., 2018). Surprisingly, general utilization of these models for operational support for decision-makers is still limited. One potential reason for this could be the high computational requirements, moving from the model building phase to operational insights is often limited by computational requirements. This is especially problematic when models are subject to high complexity, thereby requiring a large set of experimental runs. Which is, as discussed in the next section, often the case for crowd models.

2.1.2. The complexity of and the uncertainties in modelling crowds

This section discusses the complexity of and uncertainties in crowd modelling. First, cultural and location dependent elements are discussed that influence pedestrian crowd behavior. Secondly, different approaches regarding the modelling of crowd movements, or models of locomotion, are illustrated.

The influence of location on crowd modelling

Crowd behavior is dependent on the specific location that is examined. Based on video analysis of urban pedestrians, Kaminka and Fridman (2018) found four cultural factors of influence on urban pedestrian movements. These factors are supported by other works in the field, and can be seen as important factors to consider when constructing a model of a crowd. Table 2.1 summarizes these factors.

Table 2.1: Influential, culturally dependent, factors regarding pedestrian crowd modelling

Factor	Complexity	Sources
Group forming	Walking behavior of pedestrians is significantly influenced by group dynamics. Uncertainty is present in the distribution of groups and the underlying group behavior mechanisms, depending on the modelled location of interest.	(Köster et al., 2011; Moussaïd et al., 2010) and (Kaminka & Fridman, 2018; Lu et al., 2017)
Walking speed	Pedestrian walking speed can depend on numerous factors, such as the distribution of age. But also on location specific factors.	(Kaminka & Fridman, 2018; RiMEA e.V., 2016)
Side preference	Side preferences of pedestrians in regard to avoiding collisions can differ per location and culture.	(Kaminka & Fridman, 2018; Moussaid et al., 2009)
Personal space	Depending on the modelled location and culture, pedestrians strive for different sizes in personal space.	(Kaminka & Fridman, 2018; Mayr & Köster, 2021)

The first factor of interest regards the existence of pedestrians walking in groups. Results from Kaminka and Fridman (2018) show not only variations in the tendency to walk in groups as opposed to walking alone, but also in the composition and spatial formations of the groups. The existence of groups is especially of importance since their occurrence is found to significantly influence crowd dynamics (Köster et al., 2011; Lu et al., 2017). Additionally, research from Moussaïd et al. (2010) confirms the importance of groups due to their high presence in urban settings (up to 70%) and shows the significance of group formation with an important trade-off between social interaction and walking speed. Moreover, it seems likely that in specific events, group forming could potentially be even higher, since specific events can attract more pedestrians coming together rather than alone.

Secondly, the walking speed of pedestrians can differ significantly depending on the location of interest. Kaminka and Fridman (2018) illustrate the differences between walking speeds in different cultural locations. Not only do results indicate a variation in the speed regarding individual movements, the effect of groups on the speed also differs per location. Furthermore, depending on the age of pedestrians, walking speeds can vary significantly (RiMEA e.V., 2016, p. 15). A model of a crowd of relatively young pedestrians will differ significantly in walking speeds compared to an older pedestrian population. A mix between the ages is clearly the most realistic, leading to complexity regarding the speeds that individual pedestrians in a crowd model have.

The third factor entails the side preference of pedestrians. When two pedestrians cross paths at the same time, both would likely choose a side to defer to, in order to avoid collision. However, their preferred side is found to be dependent on the region they are used to (Moussaid et al., 2009). This is in line with results from Kaminka and Fridman (2018), showing significant differences between cultures. For example, the right side was found to be strongly preferred in England and Canada, whereas the left side is preferred in Iraq. More concretely referring to crowd models, such findings implicate that a crowd model of, e.g., an airport with pedestrians originating from different regions would need different behavior rules as opposed to a model of a crowd that is mostly homogenous regarding region of origin of the pedestrians.

Lastly, the preferred personal space is found to differ between cultures. Results from (Kaminka & Fridman, 2018) show that where the average personal space kept in Canada is close to 70 centimeters, in Iraq it comes close to 30 centimeters. Recent studies on the effect of social distancing on pedestrian walking behavior furthermore show the importance of the preferred personal space, where the desired social distance was found to influence the personal space more than the width of a corridor (Mayr & Köster, 2021).

Modelling pedestrian movements

In order to go from an origin to a certain destination, pedestrians have to navigate throughout the available space. However, the question on how to formalize these mechanisms best into crowd models is still an open debate. This modelling part of pedestrian movement is referred to as model of locomotion. When inspecting previous implementations of locomotion models, a wide range of different approaches can be found. Current, state-of-the-art, implementations and proposals range from simple, yet sophisticated, rules (Moussaid et al., 2011) to recently proposed complex mathematical optimization techniques (M. J. Seitz & Köster, 2012). It is exactly this diversity, without a universally accepted model of locomotion, that pedestrian dynamics framework Vadere was introduced (Kleinmeier et al., 2019). Three popular, yet distinct, locomotion approaches implemented in Vadere are discussed here. Where both advantages and disadvantages of each are illustrated.

Cellular Automata Model (CAM) is a classic approach that divides the simulation area into smaller parts or a grid. Burstedde et al. (2001) illustrate a CA approach for pedestrian dynamics where each small part, or cell, of the floor field can be occupied by exactly one pedestrian or none. Now, pedestrians can move towards one of their neighboring cells during each simulation step, given a set of specified rules. One of the main advantages of the use of CA is its simplicity and thereby the low computational needs (Kleinmeier et al., 2019). These advantages do come with certain limitations, especially regarding simulating pedestrian dynamics. The most important of which, as illustrated by Kleinmeier et al. (2019), are limited variations in step length and directions that agents take, in addition to deviated trajectories and situation in which agents get stuck. Even in recent, more advanced, applications of CA for pedestrian simulation, many of these limitations still exist (Lubaś et al., 2016; Wąs & Lubaś, 2014).

The Social Force Model (SFM) is a force-based model, in which agents are attracted or repulsed from a certain point in space by external forces (Helbing & Molnar, 1998). Helbing and Molnar (1998) illustrate that in simple form, this model can capture pedestrian motion like lane development for agents walking in the same direction and oscillatory diversions of directions in narrow areas. Many extensions on the model exist, calibrating the model even further on pedestrian behavior (Johansson et al., 2007). Kleinmeier et al. (2019) summarize the SFM and its implications regarding pedestrian simulation. They illustrate the model's advantages, such as its ability to capture pedestrian behavior. However, the authors also discuss limitations such as oscillating trajectories and overlapping agents.

The Optimal Step Model (OSM) was introduced by M. J. Seitz and Köster (2012) with the objective to combine the advantages of the previously mentioned CA and SFM without including the inherent disadvantages. OSM differs from the previously mentioned models by using a continuous space while still cohering to the natural movements of pedestrians (M. J. Seitz & Köster, 2012). After its original introduction, the model was continuously improved. As presented by Von Sivers and Köster (2015), one important element was the extension, or ability of the model, to include dynamic stride length. Often, when other models reduce pedestrian walking speeds in high density situations, the length of the strides is not considered (Von Sivers & Köster, 2015). The OSM, as implemented in Vadere (Kleinmeier et al., 2019), can, however, take this into consideration. The OSM is described to remain computationally efficient and even allows for the inclusion of complex human behavior, such as group forming (M. J. Seitz & Köster, 2012). In current model implementations, for example in the work of

Kleinmeier et al. (2019), all agents inspect their near surroundings and discretely take a step, where the length depends on the available free space, towards the direction with the highest utility. Even more so, recent advances implemented event-driven update schemes into the OSM (M. J. Seitz & Köster, 2014). This means that the model can actually handle pedestrian movement based on the real order of occurrence, instead of using unit clock time. However, as discussed in the work of Kleinmeier et al. (2019), large-scale applications of the OSM can lead to high computational needs. In order to determine the next step for all agents, the model has to solve heavy optimization functions. A large set of agents can thus be significantly computationally demanding.

2.1.3. Utilizing crowd models for in-event crowd management

Based on the current landscape of in-event crowd management, it is evident that crowd models show great potential for operational support on in-event crowd management. This way, potential in-event crowd management measures can be shaped and tested, informing hosts and local decision-makers about their potential. Most of all, models thereby help to understand the system of crowd management more thoroughly, enabling the comparison between different in-event measures before they have to be implemented in real life.

However, crowd models are currently not used to their full potential regarding operational crowd management support. That is, two apparent challenges of using crowd modelling for decision-support remain: (1) computational requirements, models in the field of crowd simulations are either very computational expensive, or lack important detailing elements inherent to microscopic crowd modelling, such as heterogeneous crowd elements (Duives et al., 2013); (2) crowd models are subject to inherent complexity and thereby a large amount of surrounding uncertainties.

A methodology is needed that can handle both a large number of uncertainties and computational requirements, all while generating fruitful insights for crowd managers. This work therefore proposes a methodology consisting of three steps: (1) exploration; (2) selection; and (3) evaluation. This methodology is specifically aimed at utilizing complex crowd models to understand the system of interest more thoroughly, and generating operational support for crowd management.

Figure 2.1 illustrates the proposed methodological steps. The first phase is targeted at exploring the model. That is, without crowd management in place, analyzing what might happen. Second, the case selection phase is targeted at reducing the number of cases to evaluate in-event crowd management measures, using insights about the system of interest generated in the first phase. Where the first stage uses a large number of samples, the second phase is targeted at reducing this number while still including different sets of circumstances. Lastly, the evaluation step is targeted at evaluating potential in-event crowd management measures over the selected circumstance in the second step, thereby reducing computational time.

In order to apply the methodology in practice, a crowd model of an actual crowd management case study has to be present. Therefore, the methodology of this work is divided into two phases: (1) constructing a crowd model of the Grote Markt; and (2) applying the proposed methodological steps for model experimentation using the constructed model. The next section discusses the theoretical context in regard to crowd modelling at the Grote Markt, and the model itself is presented in Chapter 3. The applied methodological steps for experimentation are discussed in Chapter 4.

Figure 2.1: Proposed methodological steps for model-based experimentation

2.2. Case study: In-event crowd management at the Grote Markt

In-event crowd management measures can take on many forms and can be applied in different contexts. This section discusses the crowd modelling aspects specific to the case study of in-event crowd management at the Grote Markt. In-event hereby relating to crowd management during a busy shopping day, or when a specific type of event is happening. First, a discussion follows on how to measure safety in a pedestrian crowd model of the Grote Markt. Then, main modelling uncertainties are identified with the help of previously discussed literature. Finally, three in-event measures specific to the Grote Markt are proposed: traffic regulators, directional guidance, and object placement.

2.2.1. Measuring safety

There are several potential performance indicators regarding crowd behavior in an urban context. This case study focuses on preventing situations in city centers that are classified as unsafe regarding pedestrian crowds. This section discusses two potential measures within this context: crowd density and walking speed.

The walking speed of the pedestrians throughout the shopping street is regarded as an important measuring element. If certain elements or situations would decrease this speed, this could be seen as an indication of unwanted effects. That said, this research focuses solely on pedestrian crowds without the interference of other traffic, like cars. Still, speed can be seen as an important element of safety when an impactful event takes place, and it is often an important element in evacuation studies (Shende et al., 2011; Teo et al., 2015; Von Sivers et al., 2016).

The speed value in this research is based on a calculation that considers the current position of a pedestrian and calculates the distance to the position one second earlier. This gives an estimation on the current speed. The speed calculation can thus be specified as:

$$speed = \frac{distance}{time} \tag{2.1}$$

where:

$$distance = \text{meters traveled last second}$$
$$time = 1 \text{ second}$$

Crowd density can additionally point to unsafe situations. When two pedestrians both walk towards their own destination, the chance exists that their paths cross. In any situation, when a lot of pedestrians clusters in one place, the possibility exists that they physically bump into each other or experience other physical discomfort. As illustrated in research from Zhou et al. (2021), certain threshold values can be used to consider safety related to crowd density. In this research, high crowd density is considered a possible indicator of unsafe situations.

To measure the density itself, there exists a multitude of quantification methods on crowdedness for pedestrian movements (Duives et al., 2015). This research deliberately focuses on a simplistic measurement of crowdedness, from now on referred to as density: the total count of pedestrian divided by the total area of the measurement location, similar to the described grid-based measure in the work of Duives et al. (2015). Since the mean and the maximum value of the whole simulation is considered, this type of measure is expected to not suffer much in precision, while the implementation remains easy. The density calculation can thus be specified as:

$$density = \frac{\sum pedestrians}{area} \tag{2.2}$$

where:

$$pedestrians = \text{pedestrians currently within the density measurement area}$$
$$area = \text{measurement area in square meters}$$

The measured speed in this research is seen as an indirect indicator of safety, while the density is used as a direct indicator. In order to generate an indication of safety, based on the speed and density measures, threshold values are utilized in this research. The measured density thereby directly indicates potentially unsafe situations. Based on the work of Schaap and Dijkshoorn (2020, p. 15), a density higher than one pedestrian per square meter in a shopping street is classified as potentially unsafe. In addition, the measured speed can inform about potential differences in situations where the density results in a similar unsafe or safe classification.

2.2.2. Main modelling assumptions

There are multiple important factors to consider when constructing a pedestrian crowd model of the Grote Markt. The location dependent factors (see Section 2.1.2), bound to cultural influences, and the model of locomotion (see Section 2.1.2), need to be considered with care. This section discusses the main assumptions made in regard to the crowd model of the Grote Markt. A summary of the assumptions, with the associated sources, can be found in Table 2.2

Table 2.2: Overview of assumptions

Category	Assumption	Source
Group forming	Around 70 with sizes 1-5%	(Moussaïd et al., 2010)
Walking speed	Around 1/ms	(RiMEA e.V., 2016; Weidmann, 1992) and (Sutheerakul et al., 2017; Yuan et al., 2016)
Personal space and object repulsion*	Relative strength factor	(Gödel, 2022)
Amount of pedestrians*	Wide ranges from all directions	Based on total walkable area
Model of locomotion	Optimal Step Model	(M. J. Seitz & Köster, 2012)
Side preference	No significant deviations	-
Spatial layout	-	Data from Open Street Map and Satellite imagery (Netherlands Space Office, n.d.)

*Is specified in the model building phase

Regarding group forming, two aspects are of relevance: (1) the approach on modelling groups; and (2) the number of groups and their sizes.

First, this research follows the approach on modelling groups introduced by Köster et al. (2011), which is extensively described and calibrated on empirical findings in later works (Köster et al., 2014; M. Seitz et al., 2014). Table 2.3 illustrates this approach in the context of this research. Note that this approach is already implemented and supported within the Vadere simulation framework.

Second, regarding the amount of group forming, empirical findings on the quantity of group forming and their distribution in sizes presented by Moussaïd et al. (2010) are used to estimate the presence of groups at the Grote Markt. However, since empirical studies on the Grote Markt are currently not present, a broad range for the amount of group forming will be used in the simulations. These distributions can differ significantly based on the location and circumstances. Research from Moussaïd et al. (2010) indicated a total group forming of 55% during an afternoon on a working day, and 70% during a weekend day in a commercial walkway. Both were indicated based on video recordings made in the city of Toulouse, France. Furthermore, a distribution of group sizes was determined by the researchers, showing differences mostly in the number of pedestrians that walk in pairs of two.

The walking speed of pedestrians is another important aspect regarding the crowd model of the Grote Markt. In earlier work (M. J. Seitz & Köster, 2012), a value of 1.34 meter per second is assumed as the mean for the speed distribution, based on a German publication on walking speeds of pedestrians (Weidmann, 1992). However, given the lack of empirical data on actual speeds at the Grote Markt, this research uses a range of values to account for other possible cases. Based on the work of Weidmann, RiMEA E.V. includes a distribution of walking speeds of pedestrians in its guidelines for evacuation studies (RiMEA e.V., 2016, p. 15). Depending on the age, it can be seen that the walking speeds variate significantly.

Furthermore, as pointed out in other studies (Sutheerakul et al., 2017; Yuan et al., 2016), the walking speed of pedestrians in a shopping area will likely be lower compared to speeds found during evacuations. This research therefore assumes a lower base value compared to the previously used 1.34 m/ms, a reduction of 0.34 meter per second is assumed, bringing the base value to 1.0 meter per second. A range for the uncertainty sampling with significantly lower and higher mean values is also utilized. Based on Weidmann (1992) and RiMEA e.V. (2016, p. 15), a lower bound of 1 meter per second as the average value is assumed. Again, considering the different location of application, a

Table 2.3: Approach used on modelling groups

Note. This approach is already implemented in Vadere, and introduced by (Köster et al., 2011; Köster et al., 2014; M. Seitz et al., 2014)

Group element	Approach
Leader	Have a dynamic group leader: the one closest to the destination.
Speed	Members stay close to each other and walk at the same speed. Speed will only vary for short periods of time so that the group stays together. The ones behind the leader will increase speed and the ones in front of the leader slow down.
Direction	Group members communicate and hence face the same direction.
Structure	Groups have a simple spatial structure that will remain throughout the simulation.
Separation	The group slows down when a pedestrian is far behind to avoid separation.

shopping street compared to an evacuation setting, the lower bound is assumed 0.34 meter per second lower, at 0.66 meter per second. Such a case could, for example, occur when the walking public merely consists of a mix of older and younger people (e.g., grandparents with their children). As for the upper bound, based on Weidmann (1992) and RiMEA e.V. (2016, p. 15), a value of 1.5 meter per second is assumed. Again, this bound is lowered by 0.34 meter per second to a value of 1.16 meter per second. Such a case can occur when the walking public would merely be teens and pedestrians around the age of 40 or 50 years old.

Besides the walking speed, both the personal space factor of pedestrians and the distance pedestrian keep from buildings are also of importance. The work of Gödel (2022), on influential parameter in pedestrian models of bottleneck cases using Sobol' indices (Sobol', 2001), supports the consideration of walking speed (mean and standard deviation) as a parameter of interest, next to the personal space strength factor and the object repulsion. Therefore, this study also includes both the personal space strength and object repulsion as parameter of interest. Since the same OSM model is used, the potential ranges, 5.0-50.0 (personal space) and 2.0-10.0 (object repulsion), as used in the work of Gödel (2022, p. 27), are replicated.

While the study of Gödel (2022, p. 77) deems the number of pedestrians as non-influential following the performed experiments, this number of pedestrians is considered of special interest is this study. This is due to the variation in the aim of the study. Gödel (2022, p. 26) focused on the flow of pedestrians through a bottleneck in the context of evacuation. Whereas instead, this study focuses on a shopping street, measuring the speed and densities op pedestrians. The number of pedestrians is in contrast expected to highly impact the measured densities in the shopping streets. Furthermore, due to the lack of empirical data on the Grote Markt, this study includes a broad range of possible configurations of pedestrians coming from multiple directions, based on the total walkable area. The exact amount is therefore determined in the model building phase. In addition, to remain comparability between experiments, maximum values for the number of generated pedestrians from each side of the Grote Markt are used.

This research favors the utilization of the Optimal Step Model M. J. Seitz and Köster (2012) as the model of locomotion. This choice is made due to the realistic behavior it can achieve regarding pedestrian crowd movements. For more details, see Section 2.1.2.

The side preference regarding pedestrians is considered less important regarding a crowd model of the Grote Markt. Pedestrians are expected to visit the area on a more frequent basis, and are therefore familiar with the unspoken side preference. Since large deviations of side preferences are not expected, this factor is left out of further consideration within the scope of this study.

Lastly, besides the previously named factors, the importance of the spatial layout of the Grote Markt has to be noted. The shape and layout of the building, terraces and remaining walking space, significantly influences the dynamics of the resulting model. For this work, spatial elements are based on data from Open Street Map[1] and on satellite images (Netherlands Space Office, n.d.). More details on exact spatial layout can be found in Chapter 3 and in appendix A.

2.2.3. In-event crowd management measures

This section illustrates in-event crowd management measures that can be applied in-event at the Grote Markt. That is, they can be activated when the event is happening, and scaled-up or scaled-down when needed by decision-makers. An overview can be found in Table 2.4.

Table 2.4: Overview of in-event crowd management measures

	Name	Description	Based on	Model implementation
Control 1	Traffic regulator	A person instructing pedestrians to stop and continue in another direction	(Koukopoulos et al., 2018; Park et al., 2018)	Additional model obstacles, blocking certain routes
Control 2	Directional guidance	Signs displaying alternative routes for pedestrians	(Gorrini et al., 2014; Pellegrini et al., 2009)	Varying random distributions for pedestrian targets
Control 3	Object placement	Line barriers that can be placed in walkable areas	(Karbovskii et al., 2019)	Additional model obstacles, representing line barriers in shopping streets

The first proposed measure is the use of traffic regulators. Although focused on (angry) crowds and police, elements as discussed by Park et al. (2018) could potentially be tested on regular urban crowds, under everyday conditions. In other words, traffic regulators could form a line and block pedestrian movement in a certain direction. Such regulators could monitor the area on their own and decide whether pedestrians should be instructed to wait a short amount of time before continuing their journey. Alternatively, regulators could get messages from a central control element, like in the research of Koukopoulos et al. (2018) in the context of carnival parades. This central control could monitor the crowd with sensorial data, and instruct the traffic regulators to act when necessary.

More specifically to the Grote Markt, traffic regulators could be placed at the entrances of the spaces between terraces and buildings. Often, these are very narrow and only suited for pedestrians wishing to go specifically to one of the stores or bars behind the terraces. Traffic regulators could be deployed in-event when problems are foreseen, and act as a stopping mechanism for the general pedestrian flows, as to not use the narrow corridors. In model implementation, these regulators can be simplified to obstacles, making it impossible for pedestrians to move further in certain directions. Pedestrians wanting to move towards a specific store can naturally still move past the traffic regulators in real life, as they are not a physical static object, but rather a human element that can change instructions when necessary.

The second measure is focused on directional guidance, as Pellegrini et al. (2009) found that especially destination and social interaction are crucial in crowd behavior. This insight can be combined with the finding that pedestrian walking speed slows down significantly when approaching a moving subject, for instance another pedestrian (Gorrini et al., 2014). Avoiding such collisions could speed up pedestrian flows significantly. When focusing more on urban cities in general, measures could be aimed at guiding a pedestrian throughout a crowded urban area as smooth as possible.

Applied to the Grote Markt, route guidance can be implemented with signs, instructing pedestrians to take a certain route based on insights on crowd movements. These guidance instructions can then be used to create nearly complete one direction streets, similar to already enforced one-directional measures by Dutch municipalities (NOS, 2021). In terms of model implementation, this can be realized by changing the target locations of pedestrians individually.

The third proposed measure regards the placement of objects in the pedestrian walking area. Placing objects, like demonstrated in the research of Karbovskii et al. (2019), could be enforced in-event.

[1] https://www.openstreetmap.org

Simple barriers, like traffic cones or portable fences, can be utilized when necessary by decision-makers in an urban shopping area. Karbovskii et al. (2019) found that in some situations, like in corners, t-junctions, and corridors, pedestrian flows could be enhanced by placing barriers and narrowing down space. They explain that line obstacles separate flows, thereby potentially reducing density and optimizing these flows. A notion of care should be placed, however, as Karbovskii et al. (2019, p. 7) found that, under some circumstances, it is best to not place objects within corridors. However, results indicated positive effects of line barriers in corridors with two directions of motion (Karbovskii et al., 2019, p. 5).

For the Grote Markt, line barriers in multi-directional corridors are proposed. The shopping streets at the Grote Markt can be seen as corridors, with pedestrians flowing from both directions of the street. This can be implemented within the model by placing line barrier shaped obstacles on the model area.

3

Methodology part 1: Model

The methodology of this work is divided into two parts. First, this chapter presents the constructed agent-based model. Second, in the next chapter, this model is used to assist operational crowd management using the proposed series methodology steps (see Figure 2.1). An overview of the modelling and simulation steps of this research is illustrated in Figure 3.1.

This chapter presents and discusses the constructed Vadere model, thereby answering the second research question: *"How can in-event crowd management at the Grote Markt be modelled?"*. The choice for Vadere and alternative agent-based modelling frameworks are contrasted and discussed. Afterwards, the model is presented based on the agent-based modelling steps of Van Dam et al. (2013). This includes a model conceptualization and formalization first, followed by the model implementation, verification, and validation. In addition, this chapter ends with a discussion of model stochasticity and the need for replications. A full version of the constructed model is made available on GitHub[1]. For more extensive Vadere model related details, appendix A can be consulted.

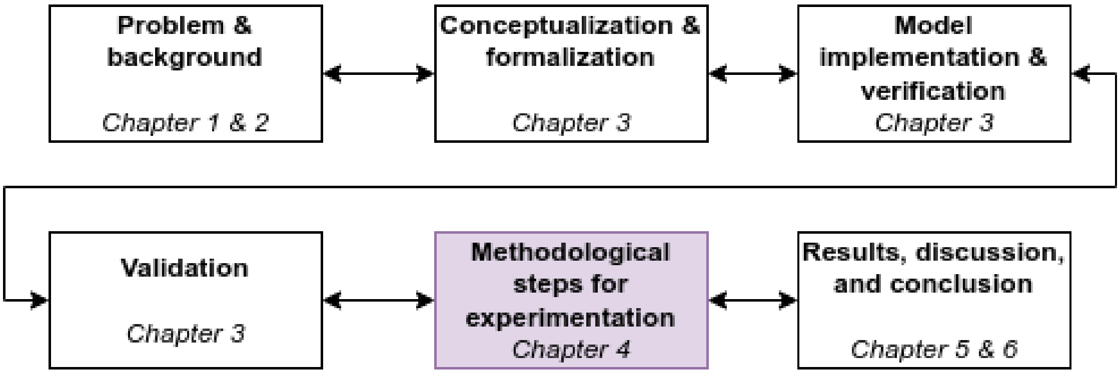


Figure 3.1: Modelling and simulation steps within this research

Note. The steps are based on the proposed practical steps in the work of Van Dam et al. (2013)

3.1. Choice of modelling framework

To integrate complex crowd behavior with heterogeneous agents and individual decision-making, an agent-based modelling (ABM) method is utilized in this research. However, various ABM frameworks exist, each with their own strong points. This section contrasts 4 open-source and distinct ABM frameworks, and argues why Vadere was chosen for this work.

[1] https://github.com/floristevito/CrowdSim

3.1.1. Comparison between open-source ABM frameworks

In the academic research domain, open-source software holds important advantages. For one, it enables faster development (Hegemann, 2022). Furthermore, open access to other work means that researchers can use and build upon previous work. Given these advantages, and to enable future work to build on this research, only open-source options are considered regarding ABM frameworks.

By means of extensive internet searches, a variety of open-source frameworks are considered: (1) Vadere, a Java-based framework made specifically for pedestrian and crowd simulation (Kleinmeier et al., 2019); (2) Cromosim, a Python library targeted at crowd simulation[2]; (3) Netlogo, a Scala and Java-based Modelling environment made for simulating multiple agents (Wilensky, 1999); (4) Mesa & Agentpy, Python libraries targeted at agent-based models in Python (Foramitti, 2021; Kazil et al., 2020).

The comparison is illustrated in Table 3.1. Six criteria that are important in the context of this work are formalized for this comparison: (1) the language of the source code, simulation engineers might be familiar with one or more languages, and prefer familiar languages to enabling faster development; (2) user interaction, where a GUI can aid fast model development alongside writing code; (3) crowd behavior support, some frameworks enable extensive out-of-the box support needed for specific kinds of models; (4) experiments, with an optional connector available to the EMA workbench connector (Kwakkel, 2017) (see Section 4.1 for more details); (5) development state, entailing active development or the lack thereof; and finally (6) documentation, guiding unfamiliar users on using the framework.

Table 3.1: Comparison between agent-based modelling frameworks

	Vadere	Chromosim	Netlogo	Mesa/Agentpy
Language (source code)	Java	Python	Scala/Java	Python
User interaction	Functional GUI + JSON based model file structure, enabling model adjustments by use of external software (Python)	Python code	Functional GUI, users can make model adjustments by writing Netlogo code, a syntax of the Lisp family	Python code
Crowd behavior support	Extensive out-of-the box crowd behavior support (e.g. multiple locomotion models and group behavior)	Some out-of-the box crowd behavior support	Limited to none out-of-the box crowd behavior support	Limited to none out-of-the box crowd behavior support
Experimentation	No connection to the EMA Workbench Some alternatives exist, but not as easy to use	No connection to the EMA Workbench	Functional and tested EMA Workbench connection	Functional and tested EMA Workbench connection
Development state	Active + easy contact with developers from Munich University of Applied Sciences	Not active, latest release dating back to April 2020.	Active	Active
Documentation	Limited and still in progress. Framework made for research purposes, not focused on extensive documentation	Limited	Extensive	Extensive

3.1.2. The Vadere simulator

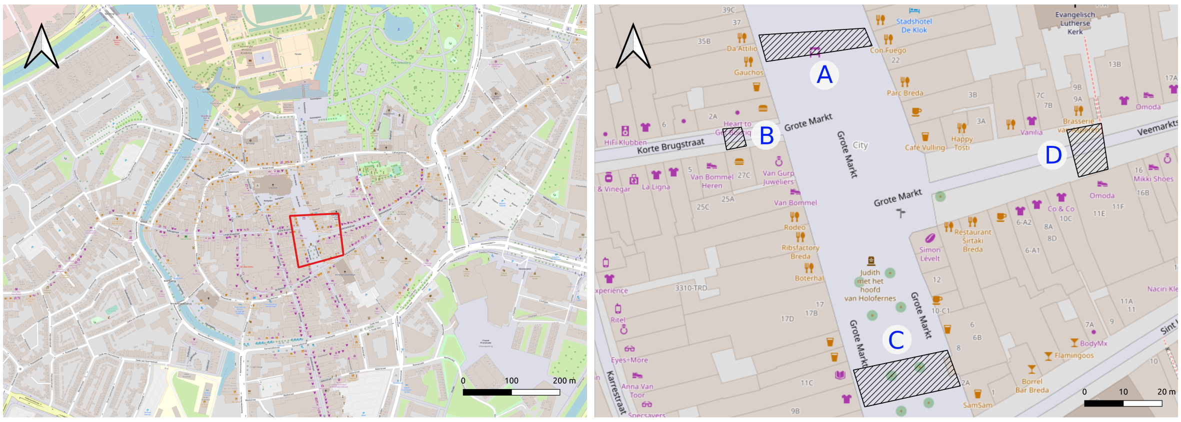
Ultimately, Vadere is considered to best serve the purpose of this research out of the four contrasted ABM frameworks. This research has a relatively short total timeline (roughly 5 months), which makes the consideration of a framework that enables fast model building crucial. However, realistic pedestrian crowd behavior on the individual, microscopic, level, is necessary in order to build a valid model. The use of Vadere enables this work to strive for both elements. The functional GUI support and JSON based model file structure, in addition to extensive out-of-the box pedestrian behavior models, make rapid yet sophisticated model development possible. That said, the model development could be realized with all four frameworks, yet it would take significantly more time with the other frameworks.

The use of Vadere in this work does, however, have three downsides that are worth noting. First, the most important one would be the unfamiliarity of the author with the framework, and its source-code in Java. Second, the frameworks' documentation is currently still limited. Third, Vadere currently has no support for the EMA Workbench, a tool that this work intends to use for experimentation. However, to overcome these challenges, this research was performed in close contact with the research group of Gerta Köster, the group behind the simulating framework Vadere, for support on the usage and technical specifications. Moreover, a new connector was built to the EMA Workbench (see Chapter 4).

[2] http://www.cromosim.fr/

3.2. Model conceptualization & formalization

This section discusses the conceptualization and formalization of the crowd model regarding the Grote Markt and is based on the system identification, concept formalization, and model formalization steps presented in the work of Van Dam et al. (2013). First, a high-level discussion on the system and conceptualization is given. Afterwards, relevant concepts are formalized and illustrated in a UML-class diagram. Lastly, the model narrative is illustrated.



(a) Model target area marked in red, located in Breda, The Netherlands (b) Detailed model area with pedestrian sources marked

Figure 3.2: Grote Markt area

Note. Basemaps provided by © OpenStreetMap contributors, https://www.openstreetmap.org

3.2.1. Conceptualization: Grote Markt, Breda

This section illustrates the model conceptualization of the Grote Markt. The model area is discussed first, followed by an overview of conceptualized model input parameters and data output. The conceptualization is concluded by a diagram that illustrates the conceptualized role of both the agents and the environment of the model.

Area of interest

The center of the Grote Markt in Breda, at the crossing of the Veemarktstraat and the Korte Brugstraat with the Grote Markt, is the area of focus of this study. Given the limited walkable area in this section of Breda, due to terraces, and the crossing of four busy shopping streets, this area is of special interest regarding safety. Figure 3.2a illustrates the location of this area within the city of Breda. This area, between the four marked locations in Figure 3.2, translates to roughly three and a half thousand square meters, based on Google Maps measurements. In addition, the streets' width at the edges of the model area are 20, 12, 25, and 4 meters for location A, B, C, and D respectively (see Figure 3.2b). However, this area is known for its dense distribution of bars and restaurants, with terraces that block the pedestrian walkways ("7x zomerse terrassen in Breda!", 2018). With the consideration of terraces (see Figure 3.7), the total walkable area is reduced by nearly 600 square meters, bringing the total to roughly 2900 square meters.

Model input

Specific to this chosen area of interest, this research considers a set of model factors of special relevance. These factors are therefore conceptualized as model input parameters. Following from the discussion in Section 2.2.2, the following variables are included: (a) the spawn frequency of all four sources; (b) the amount of group forming; (c) the speed of pedestrians; (d) the amount of strived personal space; and (e) the strength, or influence, obstacles have on the distance that pedestrians keep from them. Table 3.2 illustrates the conceptualized model parameters, with values and ranges based on the previously discussed assumptions, summarized in Table 2.2. The base values hereby represent the values used for model verification. For the final conceptualization of model input, it is important to highlight three important factors in more detail: walking speed, origin and destinations, and the number of pedestrians on the Grote Markt.

Pedestrians walk in the area of interest with a certain speed and destination in mind. Pedestrians in the model are thereby assumed to enter the crossing coming out one of four directions: the Korte Brugstraat, the Veemarktstraat, or the north or south side of the Grote Markt. These points are illustrated on the map in Figure 3.2b, and marked by A, B, C, and D. Pedestrians are assumed to have one of the three available origin destinations as their target. Furthermore, pedestrians are presumed to walk directly towards their target, and do, for example, not pause to look around. However, note that the walking speeds are expected to be lower compared to general walking speeds (Sutheerakul et al., 2017; Yuan et al., 2016), with a value ranging from 0.66 to 1.16 meter per second. Given the location, a shopping street, people are assumed to “stroll” instead of walking efficiently most of the time.

The exact number of pedestrians that walk throughout the model area is not based on exact numbers, but rather on a wide range of possible values, ranging from busy situations to more uneventful ones, estimated based on the total walkable area and the street widths at edges of the model area (see the model area specification above for more details). This range is conceptualized broadly intentionally, to include a wide set of possible circumstances. For the conceptualized area of interest, a minimum of around 350 pedestrians are assumed to have crossed the area within 100 second, while more busy circumstances are assumed to have around 1600 pedestrians having crossed the Grote Markt within this timespan of slightly more than one and a half minute.

Finally, data in regard to the exact origin and destination of pedestrians is often not gathered, or not publically available due to privacy related concerns (Autoriteit Persoonsgegevens, 2021). This research takes place in collaboration with Geodan, who has strong contacts with the municipality of Breda regarding the EU granted smart city monitoring project (Argaleo, 2021). Although this research takes place too early to use the full potential of insights generated in this project, the municipality did provide some general insights to Geodan into the origins and destinations of visitors on the Grote Markt. Figure 3.3 illustrates these insights, and Section 3.3.1 includes more details on how these insights are implemented into the model of the Grote Markt.

Table 3.2: Overview of model parameters

Vadere model parameter	Description	Unit	Range	”Base case” value
spawnFrequency(A/B/C/D)	Time between pedestrian spawn event from source location A, B, C or D.**	second	1-5	1
groupForming*	Percentage of total pedestrians walking in groups. Percentage influences the group distribution vector by changing the ratio between single pedestrians and pedestrians walking in pairs. A static value is assumed for groups of 3 (7.5%), 4 (3%), and 5 (1.5%).	percentage	50-85	70
meanFreeFlowSpeed	Mean value for the normal distribution assigning pedestrians with a freeFlowSpeed (desired speed).	m/s	0.66-1.16	1
stdFreeFlowSpeed	Standard deviation for the normal distribution assigning pedestrians with a freeFlowSpeed (desired speed).	m/s	0.15-0.30	0.26
pedPotentialHeight	Height of pedestrian potential, thereby controlling the strength of personal space for pedestrians. The higher the value, the more distance will be kept from other pedestrians.	-	5.0-50.0	50.0
obstPotentialHeight	Height of obstacle potential, thereby controlling the strength of obstacle repulsion. The higher the value, the more distance will be kept from obstacles.	-	2.0-10.0	6.0

*GroupForming determines the actual value of the groupSizeDistribution parameter in the model.
**The spawnFrequency determines the UpdateFrequency in the model and does not translate directly to the number of pedestrians, but rather to the number of spawn events. A single spawn event can translate to up to five pedestrians, with a probability depending on the set groupSizeDistribution.

Model output

This research focuses on two potential indicators of safety for pedestrian crowds: speed and density. The model is therefore conceptualized to collect both. However, the model implementation should remain efficient. Therefore, single values collected at the end of each model run indicating speed and density are preferred.

Figure 3.3: Visitors origin and destination ratio

Note. Basemap provided by © OpenStreetMap contributors, https://www.openstreetmap.org
Green arrows represent visitors coming to the Grote Markt, while red arrows highlight visitors coming from the Grote Markt. The numbers indicate the ratio between the arrows, e.g., most vistors on the Grote Markt come from the Veemarkstraat while the least amount of visitors on the Grote Markt proceed to go to the north side.

Regarding pedestrian speed, the model is conceptualized to collect the speed (see equation 2.1) of all pedestrians in the simulation at a regular frequency, every 0.4 second, and average this number. At the end of each run, the average of all collected speed values can be returned as an indication of speed.

Regarding pedestrian density, a similar approach to speed is conceptualized, where the density (see equation 2.2) is measured at 0,4 second intervals and averaged at the end of the simulation. In addition, a maximum value is returned alongside the average value, to give a deeper insight into potential peaks in density.

However, in order to get meaningful density values, isolated and static measurement areas are needed to calculate the density. This work uses four of those areas, based on experimental runs (see Section 3.3.1 and Figure 3.9). In these areas, the density can subsequently be determined by dividing the total amount of pedestrians within the area by the area size (see equation 2.2).

The role of agents and the environment

Figure 3.4 summarizes the model conceptualization, including the interaction between the agents (pedestrians) and the environment (Grote Markt). Additionally, agents also interact with each other. The diagram style is based on a similar illustration found in the work of Van Dam et al. (2013, p. 82). A clear distinction is made between the state of the agents, describing their current conditions, and the rules that take action depending on the states.

Here, each pedestrian has a current location, a destination, and a set of group members (or none). Additionally, pedestrians in the model can make an optimal step towards their target, change their destination (e.g., based on route guidance), or communicate with group members (look where they are). By considering the locations of group members while taking an optimal step, group member stay close together.

Pedestrians act with the environment, the Grote Markt, by making a step towards a certain direction or eventually reaching their destination. The environment subsequently schedules the next step of all agents and provides the available walking space. Finally, pedestrians also interact with each other. When in a dense area, pedestrians can see others nearby and slow down.

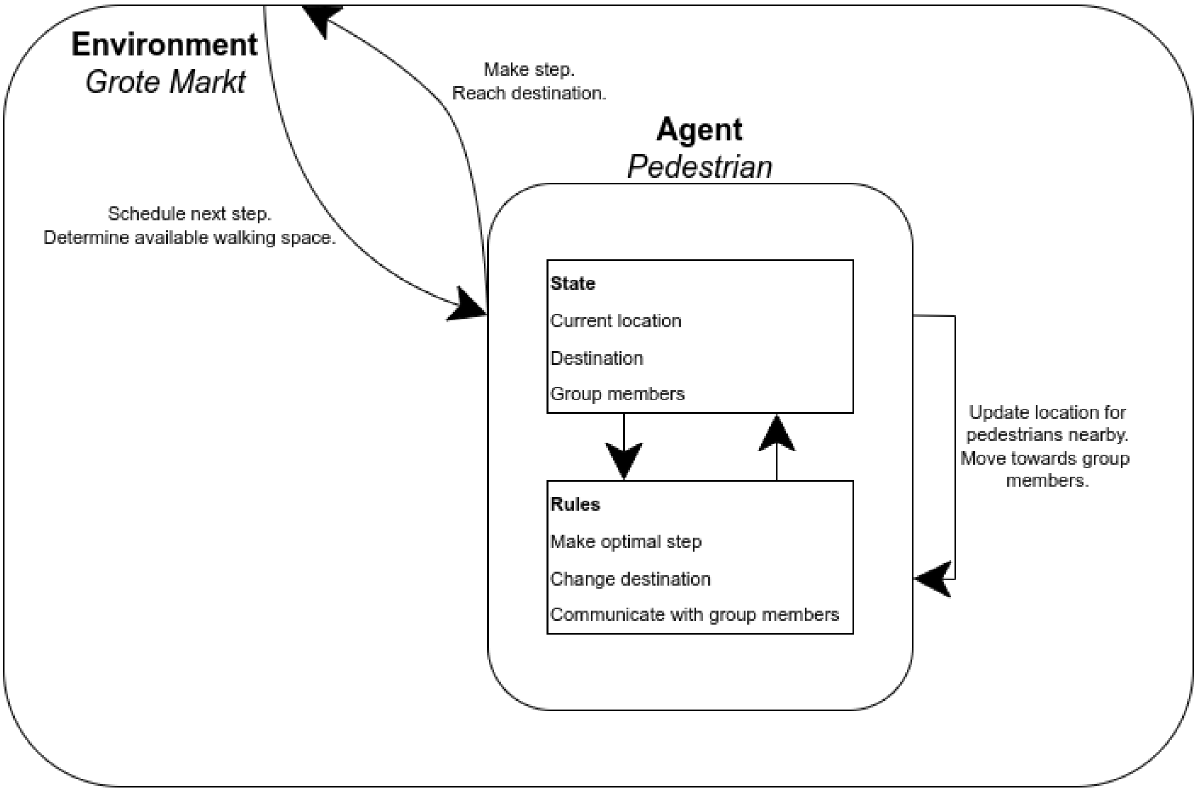


Figure 3.4: Illustration of model conceptualization

Note. Diagram style based on (Van Dam et al., 2013, p. 82)

3.2.2. Model concept formalization

Figure 3.5 highlights the formalization of relevant model concepts with the help of a UML-class diagram. The Unified Modeling Language (UML) is used, given its effectiveness in communication about ABM models, as illustrated in the work of Vermeir and Bersini (2015). The diagram (Figure 3.5) illustrates all the relevant model concepts as classes, with their own attributes and actions. The relationship between the model concepts is illustrated by the connections, of which the types are shown in the legend in the top-right corner of the diagram.

The central element of the model is the simulation model itself. This element is composed of all the important model elements: sources, targets, and obstacles. The simulationModel has a simTime, denoting the total duration of the simulation. This total maximum simulation time is set to 100 seconds. Furthermore, a seed is set, to control randomness underlying the model (e.g., the spawn location of pedestrians and their given walking speed). This seed is set to a random value in most cases. This randomness in the model is needed to account for differences in model behavior, for example in the spawn location of the pedestrians. However, the seed of the model is set to a fixed value in the manual verification, to enable repeated verification with exact reproducibility. The central model element also includes the final model statistics, as specified in Section 3.2.1, returning the densities in the four measurement areas (see equation 2.2) and the walking speed of pedestrians (see equation 2.1). The central model component includes the action updateModel, updating all model components (e.g., letting pedestrians take a step).

The central model component is composed of obstacles, representing terraces and buildings. These obstacles block the walking area of pedestrians; they have to walk around them. If no obstacles are present in a given section, pedestrians can move on the entire available space. A full overview with exact measurements of model obstacles can be found in Section 3.3.1.

Two other composing elements of the central model component are the sources and targets. Source components spawn pedestrians at the given location. The number of pedestrians that are spawned each time step can be controlled, and the origins have a unique ID by which they can be identified. The target component acts as a destination for the pedestrians, and can delete pedestrians when they

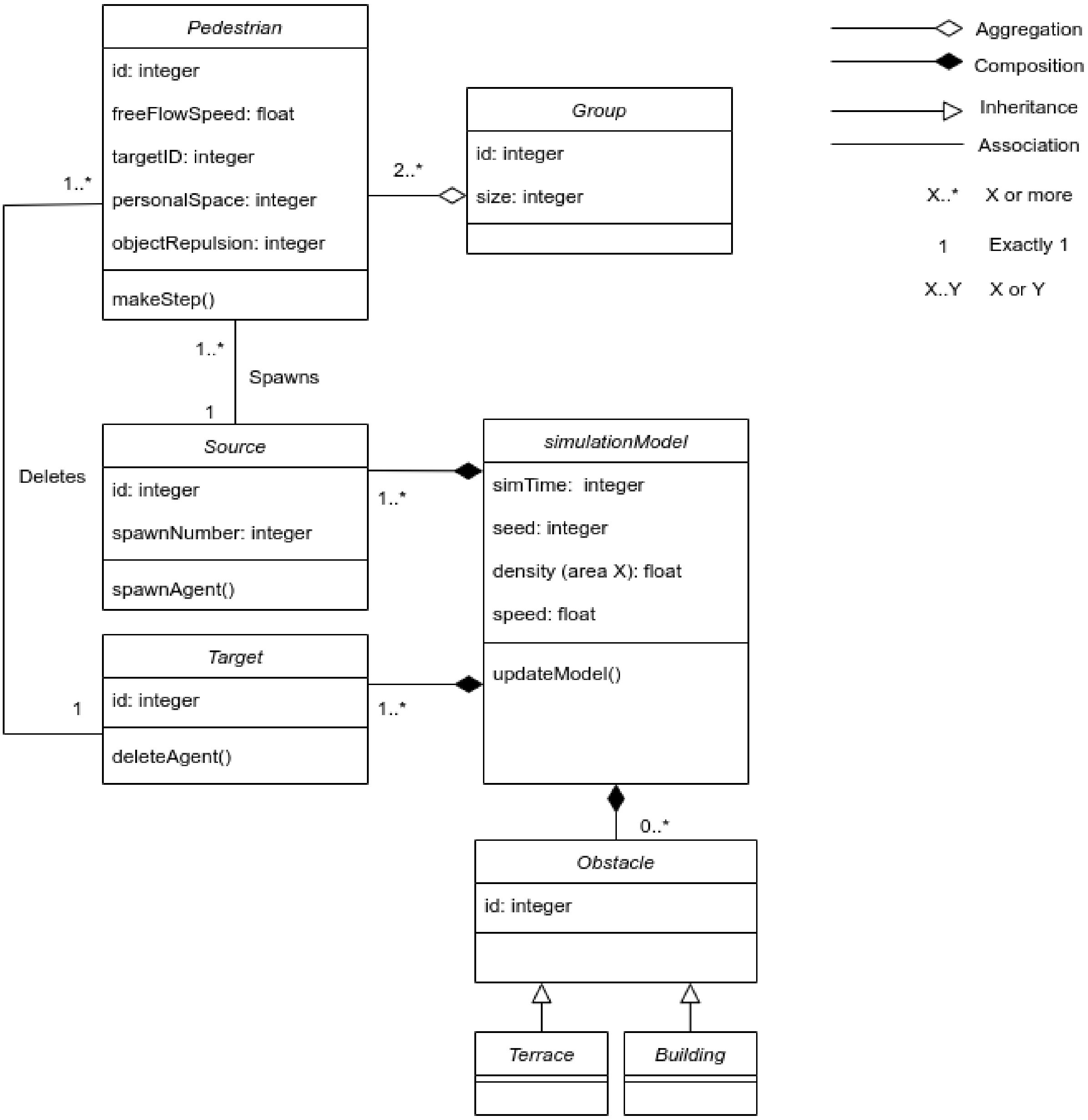


Figure 3.5: UML class diagram of the agent-based pedestrian crowd model

have reached their destination. For the case study, pedestrians are spawned from sources A, B, C, and D (see Figure 3.2). Pedestrians have subsequently one of the remaining sources as their destination, based on the origin/destination ration (see Figure 3.3.

Pedestrians that are spawned by a source component are given multiple attributes. An ID is used as identification. Their desired speed is given by the freeFlowSpeed, which is uniquely set for each pedestrian. Furthermore, pedestrians have a targetID that acts as their destination, and is linked to a target component. Besides, an attribute for both the strength of personal space and the object repulsion is included. Pedestrians also have an action that illustrates taking a step towards their destination. Finally, a pedestrian can optionally be part of a group component, linking multiple pedestrians together. These pedestrians remain close to each other. A pedestrian can thereby only be part of one group, determined at the beginning of the simulation. Thus, each group has its own id and static size during a simulation run.

3.2.3. Model narrative

To conceptualize the situation of the Grote Markt in a more formal manner, a flow chart representing the narrative of pedestrians at the Grote Markt is illustrated in Figure 3.6. This chart illustrates the steps pedestrians should take in the model. After the initial model setup, each simulation loop, a pedestrian wakes up (activates), takes information from its environment, and makes a potential move with the highest utility. Highest utility hereby meaning the movement that, depending on the target of the pedestrian and its surroundings, is optimal. The order in which pedestrians move is based on the natural order. That is, the pedestrian that finishes its step first will also be the first one to move again, also referred to as the event-driven update method (M. J. Seitz & Köster, 2014).

Note that this approach differs from the more common approach in agent-based modelling studies, where all agents move at a set (or random) order at a given time according to a regular interval. It can be argued that, especially regarding pedestrian movement modelling, the event-driven update method better captures reality, where pedestrians do not move at a set time according to a set (or random) order, but rather take a step when the previous one is finished (M. J. Seitz & Köster, 2014). Similar modelling techniques are already used extensively in the discrete-event modelling field (Schriber et al., 2015). Subsequently, when a pedestrian reaches its destination, it will be deleted from the model. In addition, new pedestrians are generated from one of the sources at a given interval frequency.

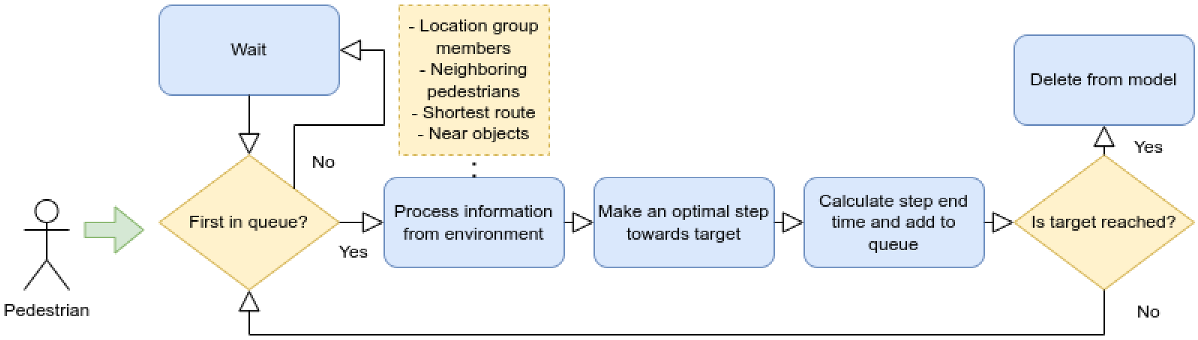


Figure 3.6: Flow chart of model narrative

3.3. Model implementation & verification

This section discusses the implementation of the model in Vadere. In addition, this section ends with the discussion on model verification, ensuring that the model is built correctly (Van Dam et al., 2013). For more details on the constructed Vadere model, appendix A can be consulted.

3.3.1. Model implementation

Model implementation is performed with the help of two of tools: (1) The Vadere GUI; and (2) Python scripts. This section covers the transformation of the model conceptualization into a Vadere model. First, general model implementation in Vadere is discussed. Afterwards, the implementation of measuring elements for density and walking speed and the proposed in-event crowd management measures are illustrated. For more technical details on the implementation in Vadere, appendix A can be consulted.

General model implementation

This general implementation of the model in the Vadere simulation framework requires certain modelling choices to be made. This section discusses eight important considerations: (1) simulation time; (2) simulation step length; (3) spatial model environment; (4) sources of pedestrians; (5) the amount of modelled pedestrians; (6) targets of pedestrians; (7) the locomotion model; and finally, (8) group behavior.

First, the Vadere model has an implemented total simulation time of 100 seconds. This choice for the total simulation duration is made considering runtime and the amount of agents that have reached their destination. Test runs with the base model configuration indicated that after 100 seconds, most pedestrians created at the beginning of the simulation have reached their destination, with the exception of those pedestrians that got stuck due to high density clusters. With the total simulation time of 100 second, the model runtime is kept reasonable for this research (within one minute).

Second, the simulation step length of 0.4 is chosen, as this is currently the default provided setting by Vadere. This setting is assumed to give the right balance between accurate results and performance in this research. The event-driven implementation (M. J. Seitz & Köster, 2014) thereby enables simulations to use the real order of occurrence instead of using set intervals with the wall clock. However, note that due to the event-driven approach, differences in simulation step length are only expected to influence the frequency at which the model updates output statistics. The working of the model itself is expected to not be influenced much by the step size, since pedestrians move in a natural order and not at random.

For the third consideration, the spatial elements regarding the model environment are set to a static value, referring to: (1) the layout of the buildings alongside the crossing; and (2) the layout of the terraces. For the first kind of spatial information, basic registration data from Open Street Map[3] is used. The second kind, the layout of the terraces, are based on satellite images (Netherlands Space Office, n.d.). Based on manual interpretation of the images, general bounds regarding size for the terraces were picked, and a distance of 1.5 meter between the terraces and the buildings is assumed. For reproducibility reasons, Python scripts are used to convert these objects to Vadere suitable (JSON) object types. Figure 3.7 illustrates the spatial layout with terraces, implemented into Vadere objects.

Figure 3.7: Spatial layout Grote Markt implemented into Vadere objects

Note. Gray area = obstacle (e.g. buildings and terraces), orange area = origin and destination points of pedestrians, purple area = terraces

[3] https://www.openstreetmap.org

Fourth, as illustrated in orange in Figure 3.7, pedestrians (agents) are spawned from one of four origin areas. These areas are known as “sources” in Vadere. Agents are modelled to be randomly placed on free spaces within the spawn area (marked in orange). To model real-world behavior as realistic as possible (see Figure 3.3), source B generates the most pedestrians with a spawn frequency of four per second. Source C follows, with three spawn events per second. Source A subsequently has two spawn events per second and source D one spawn event per second. Agents are either alone or part of a group from this moment of spawning. In the implementation of this model, the amount of spawned pedestrians and the presence of group forming can easily be changed by model input parameters.

Regarding the fifth consideration, the number of pedestrians included in the simulation relies on the spawn events. Meaning that the set “spawnFrequency” determines the spawn event number in total, but not the amount of spawned pedestrians. The latter depends on the used group sizes. To account for this in model results, based on the spawn frequency distribution (see Figure 3.3), model sources have a set maximum of 672, 504, 336, and 168 pedestrians (source D-C-A-B respectively). This gives a total amount of maximally generated pedestrians after 100 seconds of 1680.

This set pedestrian spawn maximum does mean that when a higher group forming is used, more pedestrians are generated at the beginning of the simulation. The sources are then capped by the maximum value, stopping them from generating more pedestrians. More concretely, as a minimum, a spawn event generates on average 1.68 pedestrians per second, given the lowest taken value for group forming. At most, given the highest possible group forming value, a spawn event generates on average 2.03 pedestrians per second. This means that with the highest group forming setting, sources can be stopped after 82 seconds. The last 18 seconds without newly generated pedestrians is assumed to be remedied by the higher number of previously generated pedestrians.

This maximum number of generated pedestrians needs to be updated according to changes in the spawn frequency. Meaning that when, for example, source A generates pedestrians every two seconds instead of one, the total number of pedestrians it can generate needs to be adjusted accordingly. This is achieved by dividing the maximum generation number by the spawn frequency, and rounding this number. Depending on the used spawn frequencies, other combinations can have a lower maximum value of pedestrians, but never higher. Thus, over all combinations, the total number of maximally generated pedestrians ranges from 336 pedestrians as the lowest value, to 1680 pedestrians at most in the base case.

Sixth, underneath the source areas in Figure 3.7 are “target” areas. These areas act as points of destination for the agents in the model. All agents are assigned a certain target at the moment they are spawned. Afterwards, a “targetChanger” element changes the destination of the agents randomly to one of the three target areas (excluding the fourth one, located at the same area they are spawned from). To model the real-world case as accurate as possible (see Figure 3.3), most pedestrians get target D assigned as destination, with source C, B, and A following respectively. The assumption here is made that the underlying distribution of allocated targets is always, from the respected location of destination, 60%, 33.3%, and 17% in the order of most popular destinations respectively.

Seventh, this research favors the utilization of the Optimal Step Model as the model of locomotion. This choice is made due to the realistic behavior it can achieve regarding pedestrian crowd movements. Vadere provides extensive out-of-the-box support for OSM. A template for the model can be activated, with preloaded parameters. This OSM implementation is extensively calibrated in previous work (M. Seitz et al., 2014; Von Sivers et al., 2016).

Additionally, the Vadere simulator framework comes with its own meshing algorithm, EikMesh, aimed at efficiently creating high-quality meshes, that are highly detailed in some areas while being less refined in others (Zönnchen & Köster, 2019). This meshing of the continuous space is mainly used for solving eikonal equations, needed in Vadere to find the shortest path to a destination. This innovative meshing algorithm enables faster performance in this regard, without loosing accuracy in the results (Zönnchen & Köster, 2019). Within this research, this meshing algorithm is explicitly used to generate meshes for all Vadere scenario files in advance, to significantly speed up model runs and lower the amount of needed memory. This is achieved by using the build in meshing method incorporated into the Vadere GUI, and placing these meshes in a separate cache folder alongside the model files while running the EMA Workbench based experiments.

Finally, for the eighth consideration, the Vadere simulator includes out-of-the-box support for group behavior. This behavior is enabled by including the “CentroidGroupModel” as a submodel of the Optimal Step Model, after which model attributes can be inserted. This group model implementation (see Table 2.3) is extensively calibrated already. The Vadere group model implementation is based on the mechanisms described by M. Seitz et al. (2014). However, it is important to note that the behavior of agents who get separated from group members is still under consideration. At the moment of writing, the approach of separated group members due to obstacles comes the closest to separation, as described and contrasted against backtracking and graph routing by M. Seitz et al. (2014, p. 812).

Implementation of measuring elements

Two kinds of measuring elements are considered within this work regarding the pedestrian dynamics within the Grote Markt area: the density and walking speed. Regarding the used exploratory modelling techniques, scalar model output values are preferred compared to time series, given computational efforts. Therefore, both speed and density are reported by the model as a scalar outcome, giving the average value over the model runtime. To implement this, changes to the Vadere source-code are made. These changes are committed to the original Vadere repository and are also provided on the GitHub repository of this research.

The average density over the simulation run is measured in four bounded areas within the Grote Markt crossing. The density calculation is kept simple and therefore efficient (see equation 2.2). For more details on the exact implementation of this calculation in Vadere, see Appendix A. Based on a set of ten test runs with the base case model (see Figure 3.8), four areas that are expected to be (potential) bottlenecks when it comes to density are chosen. All implemented density measurement areas are illustrated in red in Figure 3.9.

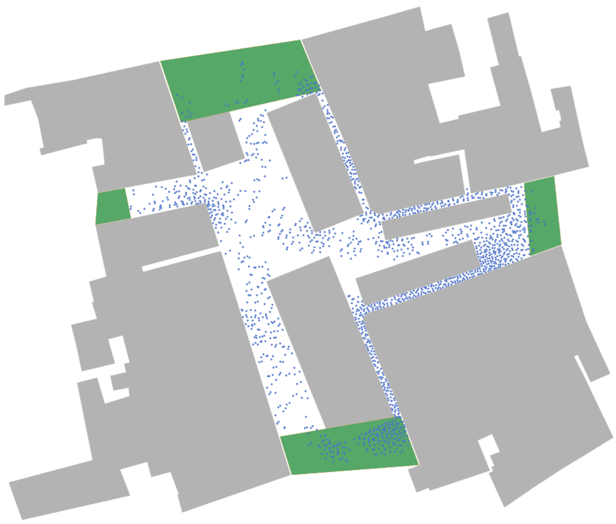


Figure 3.8: Run of the base case visualized after 90 seconds

The first area (marked by 1 in Figure 3.9) is located at the intersection of the Korte Brugstraat with the Grote Markt. In this small area, pedestrian flows from all four origins come together. Due to the central location, this area is expected to have a high pedestrian density (see Figure 3.8).

The second area (marked by 2 in Figure 3.9) is located at the center of the Grote Markt. This area is relatively large, meaning that results should always be considered with caution. This area acts as a control area. Where initial test runs (see Figure 3.8) indicated no real visible problems regarding pedestrian density, they could occur due to different circumstances caused by the uncertainty space or crowd management measure. This measurement area is therefore included to detect such potential unwanted effects.

The third and fourth area (marked by 3 and 4 respectively in Figure 3.9) are both located in the narrow space between buildings and terraces. Test runs (see Figure 3.8) indicated significantly high densities in these areas. This problem likely occurs due to pedestrians wanting to take the shortest route towards their target. However, this route, due to the presence of high density clusters of pedestrians, might not be the fastest one. It is clear that measurement areas should be monitored to potentially prevent such high levels of density. Additionally, the area directly after source D, on the right-hand side of measurement area 3, is also observed to be crowded. However, density here is likely due to bottlenecks located in measurement area 3.

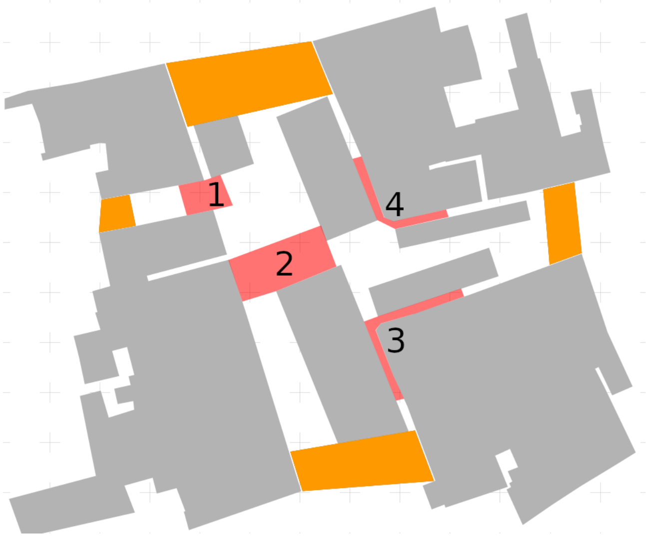


Figure 3.9: Measurement areas in Vadere

Note. Red area = measurement area, gray area = obstacle (e.g. buildings and terraces), orange area = origin and destination points of pedestrians

Additionally, the average walking speed (see equation 2.1) during the simulation run is measured over the whole area of interest. This means that for all pedestrians walking in the crossing area, from the moment they are spawned (at the beginning of the simulation), the speed is included in the average walking speed. At each interval (0.4 seconds), the average speed is reported. At the end of the simulation, these values are again averaged to one single scalar average speed outcome. This measurement area therefore is a bounded box including the whole area between the four origin points. More details can be found in Appendix A.

It is acknowledged that taking a single average value regarding the pedestrian speed might hide small deviations in actual walking speeds. Furthermore, the average value might be influenced significantly by outliers. Both the percentile and median values are therefore considered as alternatives. However, within this research, both alternatives are seen as non-satisfactory due to performance (the number of cached values should remain low) and interpretation (the average is considered the most self-explanatory to interpret). Moreover, since the speed is averaged out over all pedestrians, for each 0.4 second interval within 100 seconds, it is assumed that outliers do not affect results too much. Therefore, the average speed is considered as a pragmatic solution to measure and easily compare the speed over the large amount of model runs.

Implementation of in-event crowd management

Traffic regulators in this research illustrate a line on a certain location, blocking pedestrian movement entirely. The implementation in Vadere therefore can be reduced to a relatively simplistic polygonal object in the model area. The locations of these regulators are based on the layout of the terraces. As illustrated in Figure 3.7, the Grote Markt has many terraces blocking pedestrian movement. As a result, a relatively small walking area is present between the terraces and the buildings. Pedestrians that want to travel as fast as possible to their destination are expected to use these narrow spaces in order to optimize their route. However, this is expected to lead to potentially high densities of pedestrians in narrow areas. Therefore, the placement of regulators blocking pedestrian movements are proposed at one of the entrances of the narrow spaces between the terraces and buildings. These proposed areas are marked red in Figure 3.10. Furthermore, Figure 3.11 illustrates the object implementation into Vadere, representing a line of traffic regulators blocking pedestrian movements.

Directional guidance is implemented in the Vadere model by the use of distributions for destination values. In this research, a target is given to a pedestrian according to a set probability function

Figure 3.10: Proposed locations for traffic regulators

Note. Red area = traffic regulators, gray area = objects (e.g. buildings and terraces), orange area = origin and destination points of pedestrians

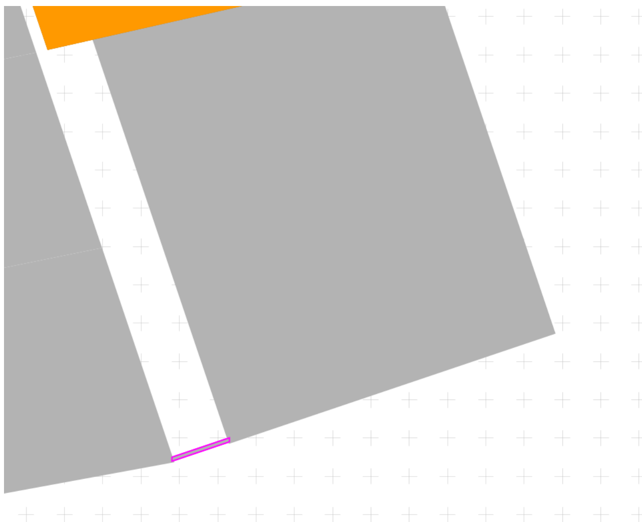


Figure 3.11: Implementation of traffic regulator obstacle in Vadere

Note. Terraces and building obstacles were merged to simplify model layout and the mesh generation process, since the areas between terraces and buildings no longer represented walking areas for pedestrians.

that represents signs instructing pedestrians to change their target and thereby route. This way, a representation of the Grote Markt area where the Grote Markt street is almost entirely one directional is proposed.

Figure 3.12 illustrates the possible pedestrian flows currently implemented into the model. As said, pedestrians get randomly assigned one of the three possible directions, with an even probability for all three. However, this research proposes to optimize flows by using directional guidance, attempting to make the Grote Markt nearly one directional without having to enforce this strictly. The colors of the arrows in Figure 3.12 illustrate the proposed guidance. Green arrows are stimulated, while red arrows are discouraged. This way, most pedestrian traffic on the Grote Markt follows one direction, from left to right. With the exception of pedestrians originating from the south side of the Grote Markt. Instead of removing this flow entirely, making comparison against the other in-event crowd management measures more difficult, it is proposed to guide pedestrians towards the Korte Brugstraat. This street is proposed given the relative low number of pedestrians originating from this direction, and the nonexistence of terraces that would potentially block the flows (see Figure 3.3 and Figure 3.7).

To encode the route guidance, newly implemented target changer objects right after the original

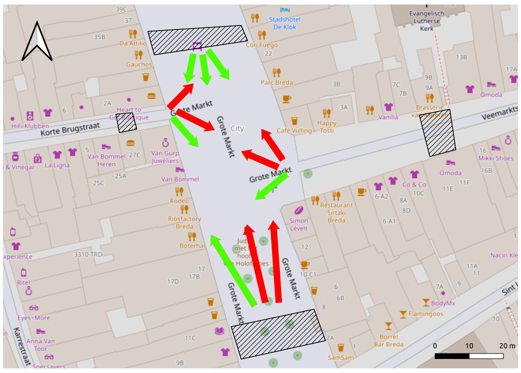


Figure 3.12: Proposed route guidance on the Grote Markt

Note. Basemap provided by © OpenStreetMap contributors, https://www.openstreetmap.org
green arrow = guided direction, red arrow = discouraged direction

sources are used. These target changers are placed in such a manner that all pedestrians originating from the source must pass over the changer object. Additionally, to ensure that opposing flows do not also change their direction when nearing their destination, the deletion distance of targets is set to 2 meters. Meaning that pedestrians are deleted just before reaching their target, making sure that the added target changer is not invoked in this case. Now, the destination of pedestrians can be changed to one of the guided directions (marked green in Figure 3.12). With the exception of the north side of the Grote Mark, where no additional target changer is needed due to the stimulation of all directions.

Regarding the compliance of pedestrians towards the given route directions, this research evaluates two different percentages of pedestrians that follow the route guidance: 100% and 25%. This enables the distinction between the cases where all pedestrians follow the guided routes, and those cases where only a small percentage complied. Furthermore, it is assumed that the directional guidance is set up in such a way that all pedestrians see the signs when they are spawned.

Figure 3.13: Proposed line barrier locations

red = line barrier locations, gray area = obstacle (e.g. buildings and terraces), orange area = origin and destination points of pedestrians

Lastly, **line barrier objects** that can be placed (and removed) in real-time in the Grote Markt area, are proposed in this research, based on the work of Karbovskii et al. (2019) on the effect of obstacles on crowd dynamics. Corridors with agents moving from both directions are proposed as locations of interest in this research. Such corridors can be identified in the Grote Markt model area in each of the four shopping streets, where pedestrians flow from both directions under normal circumstances. Figure 3.13 illustrates the proposed locations for the line barrier objects and the column, implemented in Vadere with regular obstacle polygon objects. The total of four line barriers, one in each of the shopping streets, are highlighted in red.

However, adding additional objects to the modelling area alone is not seen as sufficient within this research. Given the current model implementation, pedestrians in such case would ignore the cognitive aspects of the barriers. That is, given the shortest route, many pedestrians would remain to walk both on the left and right side of the barriers. To account for this cognitive aspects of barriers, additional sources and target changers are added to the model area. These components ensure that pedestrians walk on the right-hand side of the line barriers at all times. Given the current implementation and Vadere mechanisms, it was not possible at this moment to adjust for a portion of pedestrians that do not take the right-hand side, but rather still walk on the left.

It is important to note that the implementation of the traffic regulators can be seen as similar compared to the object placement elements. However, this research makes one crucial distinction between the two. Where traffic regulators are considered to represent staff member, object placement represents material objects. While both elements in this research can be implemented into Vadere by objects, blocking pedestrian movement entirely, this distinction changes the location of the elements. More specifically, traffic regulators are considered to still let pedestrians trough in the real-world case. This would block most pedestrian movement, but let “destination traffic” trough on occasion in order to, for example, enable pedestrians to reach buildings enclosed by terraces. Since the implemented Vadere model in this research excludes pedestrians with these buildings as targets, the traffic regulators can be implemented as objects blocking all pedestrians. However, the interpretation of the traffic regulators still differs from object placements. Both are therefore implemented on different locations in the model and serve distinct purposes.

3.3.2. Verification

In the work of Van Dam et al. (2013), a clear distinction is made between model verification and model validation. Where model verification targeted at answering the question of building the right thing (model), model verification regards the inquiry on the correct building of the thing (model) (Van Dam et al., 2013, p. 98). This research uses the same distinction. This section focused on model verification. More information on model validation can be found in Section 3.4.

Based on the work of Van Dam et al. (2013), four model verification techniques relevant to the domain of agent-based modelling are discussed: recording and tracking agent behavior, single-agent testing, interaction testing (minimal model), and multi-agent testing. It is important to note that the active research group behind the Vadere modelling framework already put a lot of effort, and is still continuing to do so, on the correct implementation of core components of Vadere. That is, extensive unit tests and continuous integration and deployment are already in place (Kleinmeier et al., 2019). Implemented unit tests cover the core components of Vadere and the correct working thereof. Furthermore, by using a so-called test pipeline, continuous integration further reduces errors in model implementation. For this reason, model verification in this work focuses on elements that are specific to the model application on the case study of the Grote Markt.

Recording and tracking agent behavior

States of individual agents in the model were inspected by the use of the GUI. In the constructed model, multiple scenario files can be found. A Vadere scenario file includes all information regarding the specific simulation. Here, a base scenario file and one for all three investigated in-event crowd management measures are included. However, for all scenario file configurations, two versions are included. One marked “visual” and one marked “data” (e.g., “baseCaseVisual”, “baseCaseData”). As of such, a distinction is made on the collected data output (or data processors in Vadere). Where the data variant is set up to collect only relevant data for the experimental use, optimizing performance, the visual variant is constructed specifically for visual inspection. In the visual variant, data is gathered to enable extensive visual output inspection using the post visualization tools present in the Vadere GUI. For all agents, information was collected on their trajectories and states (such as group members, speed). This way, strange or incorrect agent behavior is detected to verify model implementation.

Additionally, the Vadere source code and additions thereof were inspected with a debugger. This research utilized the integrated debugger of the Jetbreans IDEA environment. This debugger was used to detect and further investigate unexpected behavior found in the visual inspection. For example, by using (conditional) breakpoints in the Vadere source code, the illustrated disappearing agent problem was debugged to its root cause (see 3.3.2).

Single-agent testing & interaction testing (minimal model)

To explore the behavior of modelled agents further, a set of tests were used in this research. These tests were performed visually, by use of the discussed visual recording and tracking methods, and are in performed in addition to the defined unit tests within the Vadere framework. Based on the discussed single-agent testing and interaction testing in the work of Van Dam et al. (2013), these tests were performed on a model with only one agent, and on a minimal model including all the interactions and including all types of agents. For this latter model, all pedestrian group sizes were included: one single agent, one group of two, one group of three, one group of four, and one group of five. These two model configurations are included in the model files, and are named “singleAgentTestVisual” and “interactionTestVisual”. These tests were intended to cover the most important expected behavior as discussed in Section 3.2, and additionally are targeted to test if agent behavior “breaks” under more extreme conditions. Table 3.3 summarizes the test results.

The first test regards making an optimal step. Pedestrians were expected to make a step in the model given their destination and their environmental surroundings. Pedestrians should avoid obstacles, keep some distance from them, and move towards their destination in the shortest way possible. This means that they should calculate the shortest path to their destination, and not the shortest distance (trough objects). This expected behavior is confirmed in the used model.

Second, pedestrians were expected to change their destination when they are on a so-called target changer. When pedestrians are in the area of the target changer, they should update their destination. Additionally, pedestrians are expected to only update their destination once per target changer. That is, when multiple steps are made within the area of one target changer, they are expected to only change their target once, given the set of available new targets and their probabilities. This test is confirmed.

Third, pedestrians were to communicate with group members. That is, they were expected to follow their group towards the same goal. When the target is changed of the group, this should apply for all members. When separated more than sixteen meters due to objects, pedestrians were expected to split up. What is more, group members who fall behind should speed up while members in front should slow down. Lastly, they should also move in a basic structure that remains unchanged, except for temporal changes due to, for example, obstacles or crowds.

This third test is partly confirmed. Upon testing, it was noticed that group members who are too far in front slow down so much that they get stuck in the event queue. To account for this behavior, a maximum step duration of ten seconds is set. That means that when group members get so far separated from the rest of the group that their desired speed is near zero, these pedestrians will wait for ten seconds and then make a step. Afterwards, they can again wait for ten seconds if needed. This value of ten seconds is based on an estimate using the pedestrian walking speeds in the model. That is, pedestrians who are more than sixteen meters in front of the last group member, with an average walking speed of one meter per second, will have to wait at least sixteen seconds for this last group member. This value is, however, set to ten seconds, since it is hypothesized that after ten seconds, group members who are less behind than the last one might have caught up. In addition, group members left behind can temporarily speed up.

Multi-agent testing

As a last verification step, the whole model (base case) was verified visually using the previously discussed recording and tracking of agent behavior. For this, the base case of the model is used, to monitor behavior over the whole timeline, similar to timeline sanity (Van Dam et al., 2013, p. 104).

Overall, correct behavior was observed, except for the occasional disappearance of pedestrians in the model. This had to do with the event queue of the event-driven implementation (M. J. Seitz & Köster, 2014). Pedestrians who were separated from group members, thereby invoking waiting behavior, now determined a low value of their desired speed, subsequently giving those agents a very high value of their next step. These values were so high that those agents were put at such a position in the event queue that their next movement was never reached within the simulation time. In collaboration with the research group behind Vadere, this problem was subsequently fixed by the previously mentioned maximum step duration specification, and some source-code adjustments made by Vadere developers regarding the working of the group models when objects between group members were present.

Table 3.3: Overview of visual verification tests

Test	Confirmed?	Remarks
Making an optimal step	Yes	Pedestrians make logical steps, depending on destination, objects, and surrounding pedestrians.
Changing destinations	Yes	Pedestrians change targets as expected.
Communication with group members	Partly	Pedestrians far in front of other group members can get stuck in the event queue. **Solved** with maximum step duration of 10 seconds.

In conclusion, together with the implemented Vadere unit tests and the test discussed above, this research trusts the overall correct implementation of the model in the context of crowd behavior at The Grote Markt. More on model variability (Van Dam et al., 2013, p. 103) and replications can be found in Section 3.5.

3.4. Validation & sensitivity analysis

Model validation in an exploratory modelling approach differs from model validation in consolidative modelling (Bankes, 1993). Traditionally, model validation would entail comparing model outcomes with real-world data to confirm an actual representation of the real-world (Sargent, 2010). However, within this research, model outcomes of the Grote Markt are difficult to directly compare to real-world data and therefore exploratory modelling—with a wide range of possible cases—is utilized. This research defines the validation of the model as the ability of the model to be useful and convincing to explain how the system might behave (Van Dam et al., 2013, p. 127).

Model validation in this research is split into two distinct phases: microvalidation and macrovalidation. This separate validation of the two model components is also referred to as cross-validation (Moss & Edmonds, 2005). Where the micro and macro levels can be defined as:

> “The microbehavior is the behavior of observed actors and is described by autonomous software modules called agents. The macrobehavior is the behavior of a social institution (organization, community, set of customers, etc.) or a collection of such institutions and is described by the properties of the model containing the agents. The properties of the model as a whole are amenable to summary using descriptive statistics, while the behavior of the individual agents can (and we argue should) be described qualitatively.” Moss and Edmonds (2005, p. 1097)

In line with the view of Moss and Edmonds (2005), microvalidation is performed with qualitative data. Observed pedestrian behavior is compared to findings from literature, with the use of empirical backing where possible. Additionally, for validation on the macro level, properties of the whole model are investigated, such as the pedestrian density and speed values. Within this research, SOBOL (Sobol’, 2001), an extensive global sensitivity analysis that enables deep insights into the sensitivity of model outcomes of input parameters, even under non-linearity, is used for this purpose. Additionally, modelled speed and density outcomes are compared to real-world observations.

3.4.1. Microvalidation of pedestrian behavior

This section discusses the validation of model behavior at the level of individual pedestrians (agents), performed with qualitative data. That is, modelled behavior of pedestrians was compared to empirically based descriptions about how pedestrians should act. It is, however, foremost important to note that the current Vadere implementation, using extensive development pipelines, already performs a wide set of test cases on the implemented simulator (Kleinmeier et al., 2019). Part of this set of tests are the fifteen test cases regarding evacuation modelling, defined by RiMEA e.V. (2016). Additionally, other use cases with different parameter settings are also performed, and at least 49 test runs are executed for the OSM model (Kleinmeier et al., 2019). It would thus be a fruitless use of time to reproduce all the test cases laid out by RiMEA e.V. (2016), since their working is already confirmed for the implemented simulator. This research therefore uses two additional microvalidation test cases, based on the described cases by (RiMEA e.V., 2016) and proposed validation tests for group models from Köster et al. (2014): (1) a test regarding visual group behavior; and (2) a test regrading the effect of terrace bottlenecks.

Table 3.4: Requirements microvalidation of group behavior

Requirement	Confirmed?	Remarks
Group members stay together	Yes	Pedestrians in the same group stay relatively close together
Group members walk side to side when no obstacles are present	Yes	Clear side to side layout of group members visible
Faster groups overtake slower groups	Yes	The green group passes the orange group at the end of the test
Groups collectively avoid a column or splits up when confronted with the obstacles	Yes	The orange group splits up just before the obstacle, while the green group takes one side collectively
Group members who split up reunite after they passed the obstacle	Yes	The orange group reunites after the obstacle is passed

Visual group behavior

The first microvalidation test utilized within this research regards pedestrian group behavior. As discussed, based on the work of Moussaïd et al. (2010) (also see Table 3.2), group sizes in the modelled shopping street range from one to five. Köster et al. (2014) refer to these groups as "small" groups, and propose a visual validation test regarding their behavior (Köster et al., 2014, pp. 1055–1056). This validation test was explicitly reproduced within this research, specifically applied to the modelled case of the Grote Markt. Based on the proposed setup by Köster et al. (2014, p. 1055), five groups (one of each group size) were modelled to walk from the north side to the south side of the Grote Markt. Due to the wide street size, three columns of one by one meter were placed in the model area to test group separation. These obstacles could be seen by pedestrians, and subsequently cause the separation of group members.

Test results indicated correct behavior regarding all the listed requirements by Köster et al. (2014, p. 1055). Figure 3.14 illustrates this microvalidation test visually, and detailed test results can be found in Table 3.4. The used Vadere model configuration scenario file is included in the published model version of this research, as the "microvalidationGroups" scenario file.

Effect of terrace bottlenecks

RiMEA e.V. (2016) illustrates a validation test targeted at bottleneck cases, referred to as test 12. This test is naturally included in the Vadere pipeline (Kleinmeier et al., 2019). However, since this test originally is targeted at evacuation cases, it was repeated in this work, targeted more specifically for urban shopping street areas. In this test, 150 pedestrians were modeled originating from the Veemarktstraat, with the south side of the Grote Markt as their destination (see Figure 3.2b and 3.7).

This test indicated results that were as expected. That is, congestion was expected due to the presence of a terrace, significantly narrowing the movement area of the shorted path to the destination. However, this terrace ends just before the Veemarktstraat intersects with the Grote Markt. After which a new terrace, now located at the south side of the Grote Markt, begins. Based on RiMEA e.V. (2016), it was expected that the main congestions will be located before the first terraces and not in the open space between the two subsequent terraces. Figure 3.15 illustrates this test, with the corresponding Vadere scenario file included as "microvalidationBottleneck".

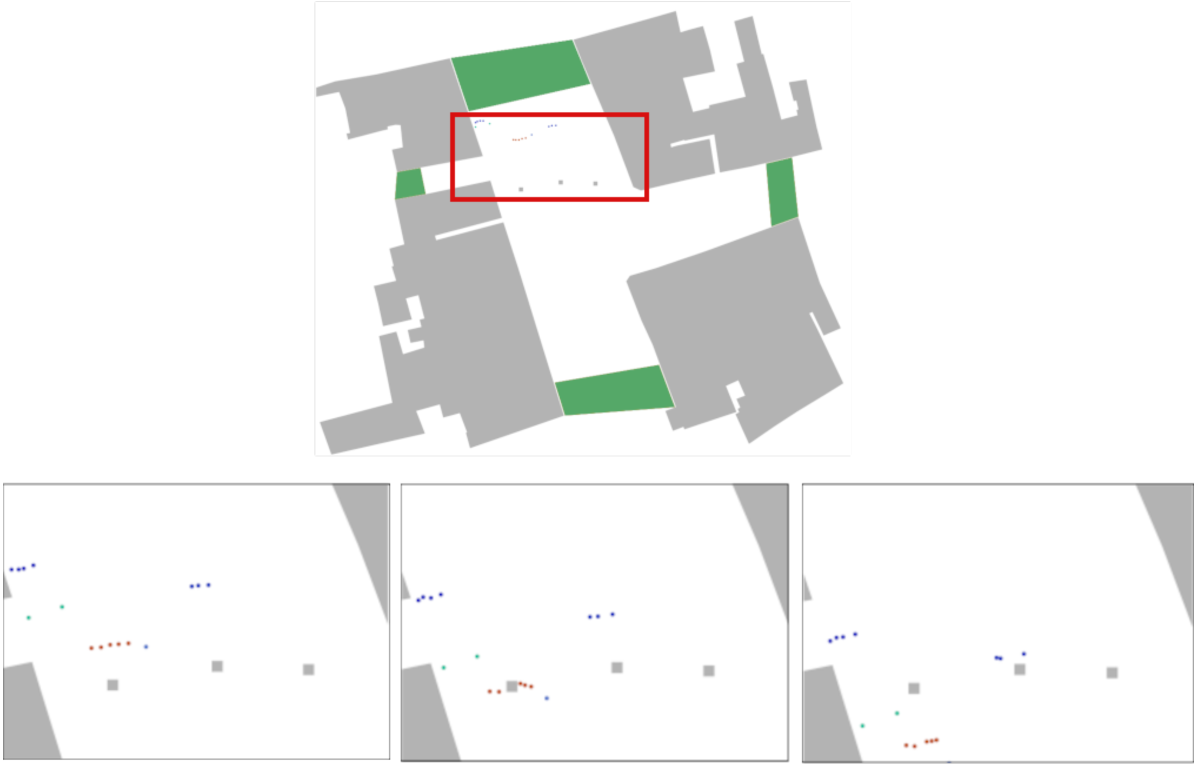


Figure 3.14: Microvalidation of group behavior

Note. Pedestrians are color coded by group

Figure 3.15: Microvalidation of bottlenecks due to terraces

3.4.2. Sensitivity analysis

A global sensitivity analysis, with all nine model parameters (see Table 3.2) included, was performed using a SOBOl based sensitivity analysis (Sobol’, 2001). This method of sensitivity analysis requires a large set of model runs. By including nine parameters, and with the assumption of 1000 cases to cover the whole uncertainty space (as used in 5.1.1), this research needed 20000 runs in order to perform the analysis, excluding replications. To remain computationally feasible, the number of replications used for this analysis was therefore reduced to 10 instead of 60 replications per case. As seen in the convergence tests, see Section 3.5, 60 replications would be ideal to average out model stochasticity. However, it can be observed that 10 replications deliver a fairly good balance between computation time and remaining variability in model results. It should, however, be noted that performing the SOBOL sensitivity analysis with this reduced set of replications could lead to slightly more variability in the outcomes.

Figure 3.16 illustrates the found SOBOL indices for the pedestrian speed and maximum density values. S1 indices denote the first order effects, thus the variance in the output value caused by the given parameter. ST indices illustrate the same, but include all the interaction effects. The focus here lies on the S1 indices, but when the ST indices highlight very small total effects, this can be an indication to discard the given outputs. Overall, it was found that SOBOL showed stabilized indices with a sample size of 1000. Additionally, the confidence intervals are also illustrated by vertical lines in Figure 3.16. The conclusion here can be made that the model is mostly sensitive to changes in the number of spawn frequencies, denoting the number of pedestrians in the simulation. This is mostly apparent for the maximum density outcomes. Regarding the found mean speed, the free flow speed is the most of the influence. These established influencing factors are logical, and overall aid to the validity of the constructed model.

3.4.3. Macrovalidation of pedestrian density and speed

Table 3.5 illustrates modelled density and speed outcomes, acquired from the total set of 1000 cases sampled over the whole uncertainty space (see Table 3.2). These values were subsequently compared to real-world data. It is, however, important to note that these values were generated over a large set of cases, without making any claims on which cases would be most frequent or likely in the real-world. Even more so, high numbers for the number of pedestrians are used. Hence, the average values found over the total set of cases, for example, do not represent the average value in the real-world system. Rather, these descriptive values illustrate the ability of the model to correctly capture real-world behavior. That said, real-world empirical data on pedestrian shopping street properties, such as densities and speed, are currently sparse. This part of model validation mainly relies on statements made by (Schaap & Dijkshoorn, 2020) regarding pedestrian crowds in shopping areas.

Table 3.5: Modelled density and speed outcomes

	mean speed [m/s]	mean density area 1 [#/m²]	max density area 1 [#/m²]	mean density area 2 [#/m²]	max density area 2 [#/m²]	mean density area 3 [#/m²]	max density area 3 [#/m²]	mean density area 4 [#/m²]	max density area 4 [#/m²]
mean	0.64	0.15	0.38	0.38	0.17	0.53	1.15	0.36	0.99
std	0.10	0.12	0.12	0.23	0.07	0.31	0.68	0.20	0.57
min	0.42	0.02	0.02	0.10	0.05	0.13	0.26	0.09	0.23
25%	0.56	0.06	0.06	0.20	0.11	0.28	0.62	0.20	0.50
50%	0.63	0.11	0.11	0.31	0.17	0.44	1.03	0.33	0.87
75%	0.71	0.18	0.18	0.46	0.21	0.73	1.58	0.48	1.36
max	0.94	0.52	0.52	1.19	0.46	1.79	3.46	1.29	3.57

Firstly, observed modelled mean speed of pedestrians were lower compared to the free flow speed (see Table 3.2). This behavior, however, is logical, since a case with a large number of pedestrians is simulated. High density is expected, slowing down pedestrians. The found mean value over the 1000 cases of 0.64 meter per second translates to roughly 2.3 kilometers per hour (note that this value represents the average value over 100 simulated time steps for all pedestrians). According to (Schaap & Dijkshoorn, 2020), average European citizens have an unhindered average walking speed of 1.34 meter per second. The authors, however, emphasize on the unhindered aspect of this speed. In a shopping street, this speed is easily reduced to one meter per second, due to circumstances such as stairs, opposing flows, or street furniture (Schaap & Dijkshoorn, 2020). In best cases, illustrated by the maximum value of mean speed in Table 3.5, modelled speed comes very close to this stated value of one meter per second. While other cases often report a lower walking speed, this is expected due to extreme circumstances used in this research. This research focuses on a busy shopping day, with many pedestrians in the streets. During almost the entire modelling runtime, most pedestrians can likely not utilize their normal walking speed. Thus, lower values for the walking speeds were expected.

Regarding pedestrian density in shopping streets, Schaap and Dijkshoorn (2020) state that values higher than one pedestrian per square meter can already lead to safety concerns. Observed modelled densities seem in line with this statement, where this value is only reached in extreme circumstances. Most mean and maximum density values were below one, in 75% of the cases. However, measurement area 3, and in less terms area 4, both do seem to cross this maximum value often. Given their location, the narrow spaces between buildings and terraces with multi-directional flows (see Figure 3.8 and Figure 3.9), these values were expected.

3.4.4. Validation conclusion

The combination of the results found from these three tests: literature based microvalidation, SOBOL sensitivity analysis (macrovalidation), and empirical comparison of density and speed outcomes (macrovalidation), demonstrated the usefulness and convincingness of the model to explain possible system outcomes. The model is therefore marked as fit for purpose: to help understand the pedestrian crowd system at the Grote Markt more thoroughly.

3.5. Model replications

The implemented model in Vadere includes stochastic terms. That is, one model run, although with the exact same model parameters, leads to varying results due to randomness in the model initialization and simulation mechanisms. The work of Gödel (2022) accordingly discusses these terms regarding the OSM implemented in Vadere. This section first highlights the most important stochastic terms introduced by this work's model implementation, after which the approach on handling this model stochasticity, and the number of replications needed for the convergence of model results, is discussed

Stochastic terms in the implemented model

One important stochastic term has regard to the speed of pedestrians. At the beginning of the simulation, all pedestrians get assigned their own free flow speed. This creates a heterogeneous population, one of the advantages of utilizing models within an agent-based modelling framework (Wilensky & Rand, 2015, p. 32). However, the assignment of these speeds to pedestrians in the implemented model is stochastic, giving different (free flow) speed values to pedestrians on each new model initialization. This speed subsequently influences the step length of pedestrians, which in itself is also a stochastic term due to the standard deviation in step length (Gödel, 2022, p. 33).

Another significant stochastic factor is the placement of pedestrians upon spawning. In the implemented model, pedestrians are spawned randomly within the source area at a given time interval. Now, each iteration, pedestrians get a possibly different position. Moreover, pedestrians are only set to spawn on free space. Consequently, when no free space is available at a given time interval, these agents are not spawned directly. Thus, depending on the spawn location initialization, more or less pedestrians can be present in the model at different locations.

Furthermore, the destinations of the pedestrians are modelled stochastically. Upon spawning, pedestrians are given a destination by a target change element according to a random draw according to set probability values. Therefore, the exact destination of each of the pedestrians can differ in each model iteration.

As explained by Gödel (2022, p. 33), the simulation loop of the OSM implementation in Vadere additionally introduces stochastic terms. For the model implemented in this work, this implies that the starting point of the optimization regarding the optimal step for pedestrians is varied each iteration of the simulation loop, with a randomly drawn offset.

Dealing with model stochasticity

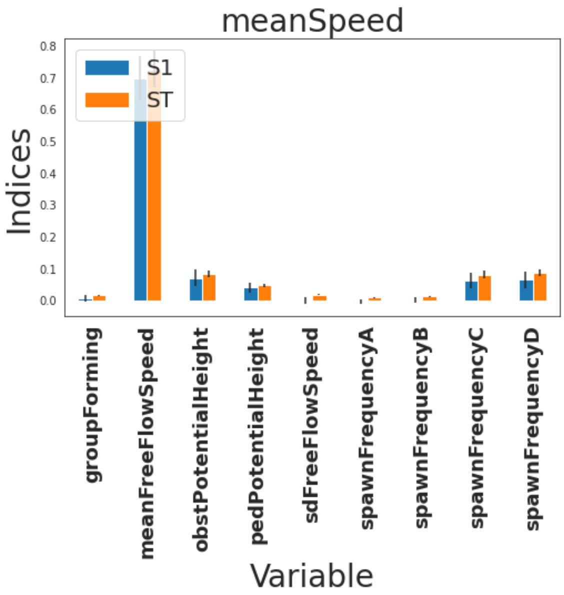
There are multiple ways to deal with model stochasticity. In the work of Gödel (2022), both (deterministic) surrogate models and model replications are used. Building or choosing deterministic surrogate models is complex and time-consuming (Gödel, 2022, p. 35). Therefore, this research utilizes the approach of averaging simulation results over a set number of model runs, referred to as model replications.

Nonetheless, it has to be noted that this approach also does not come without its disadvantages. First and foremost, performing many model replications for each combination in the uncertainty space can be demanding, especially when the uncertainty space is large. Furthermore, averaging model results could potentially lead to masking model volatility (Moss & Edmonds, 2005). However, episodes of model volatility in this research are assumed to be mainly due to stochastic terms in the model, and therefore are seen as model artifacts and not on their own representative of real-world behavior. Meaning that model results are averaged out over multiple replications in this research, but with the notion of caution on the potential hiding of model volatility.

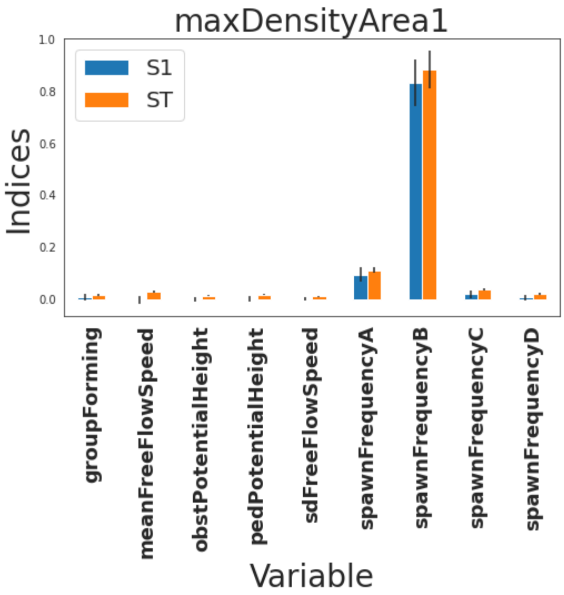
With the implementation of model replication comes the decision on the number of runs per case (combination of input parameters). Within this research, the convergence of the mean pedestrian speed is analyzed together with the mean and maximum density of area 1 (see Figure 3.9). Both the average and standard deviation of the different sample sizes are analyzed. The other three density areas are excluded, due to computational considerations and the assumption that when the density converges for area 1, it will likely also be the case for the remaining areas.

Two cases are analyzed: the base case (see Table 3.2), and a hypothesized “bad” case. This “bad” case includes all the base case values (see Table 3.2), but uses the lowest bound value for the mean free flow speed and highest bound value for the obstacle potential height. These two parameters are assumed to illustrate a “bad case” case. That is, pedestrians having lower free flow speeds while also trying to keep a great distance from obstacles (thereby indirectly reducing the available walking space). Based on the results found, illustrated in Figure 3.17 and 3.18, a number of 60 replications is assumed to be suitable within this work. Further reference on model results thus always refer to the average of model results over these replications.

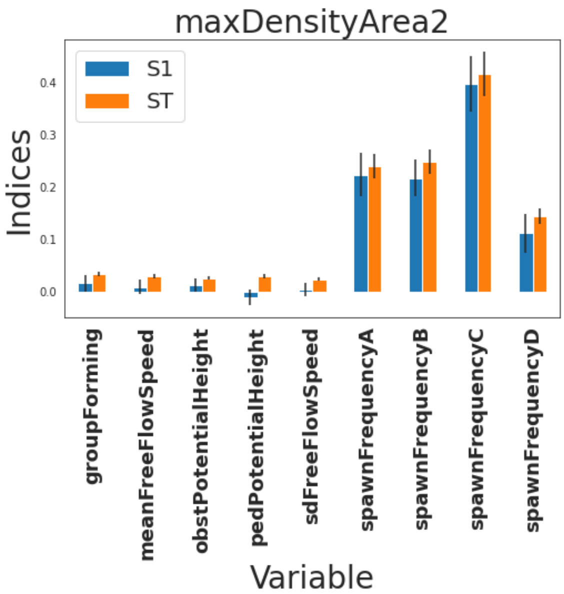
As an additional test of confirmation, 10 new runs with 60 replications each were performed (600 model runs in total). Both the base case and the hypothesis bad case showed stabilized results. The 10 resulting values for the mean speed, each averaged over 60 runs, showed a standard deviation smaller than 0.002 meter per second in the base case and approximately 0.001 meter per second in the hypothesized bad case. Both the mean and maximum values for density, averaged over 60 runs, subsequently resulted in a standard deviation of approximately 0.009 and 0.03 pedestrians per square meter respectively in the base case. In the hypothesized bad case, resulting standard deviations over the two outcomes were found to be even smaller. Therefore, confidence is given to the significantly small variability in model outcomes, given the averaging of model outcomes over 60 replications.



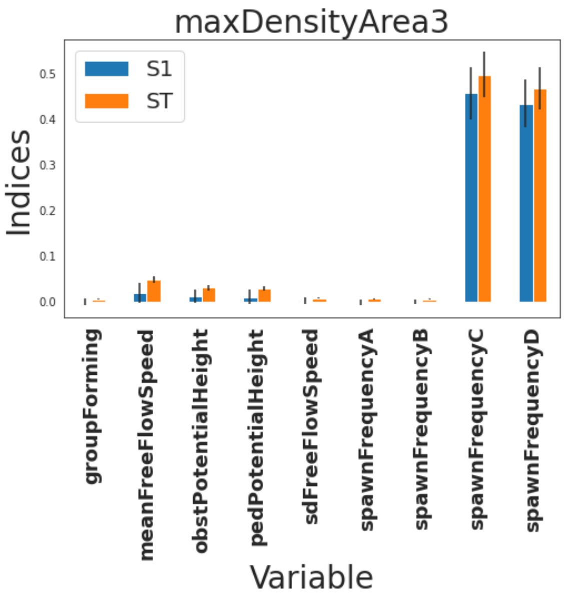
(a) Mean speed



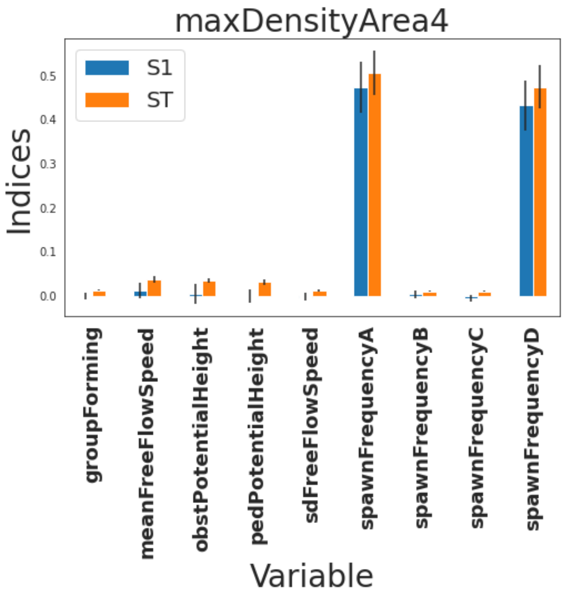
(b) Maximum density area 1



(c) Maximum density area 2



(d) Maximum density area 3



(e) Maximum density area 4

Figure 3.16: SOBOL S1 and ST indices

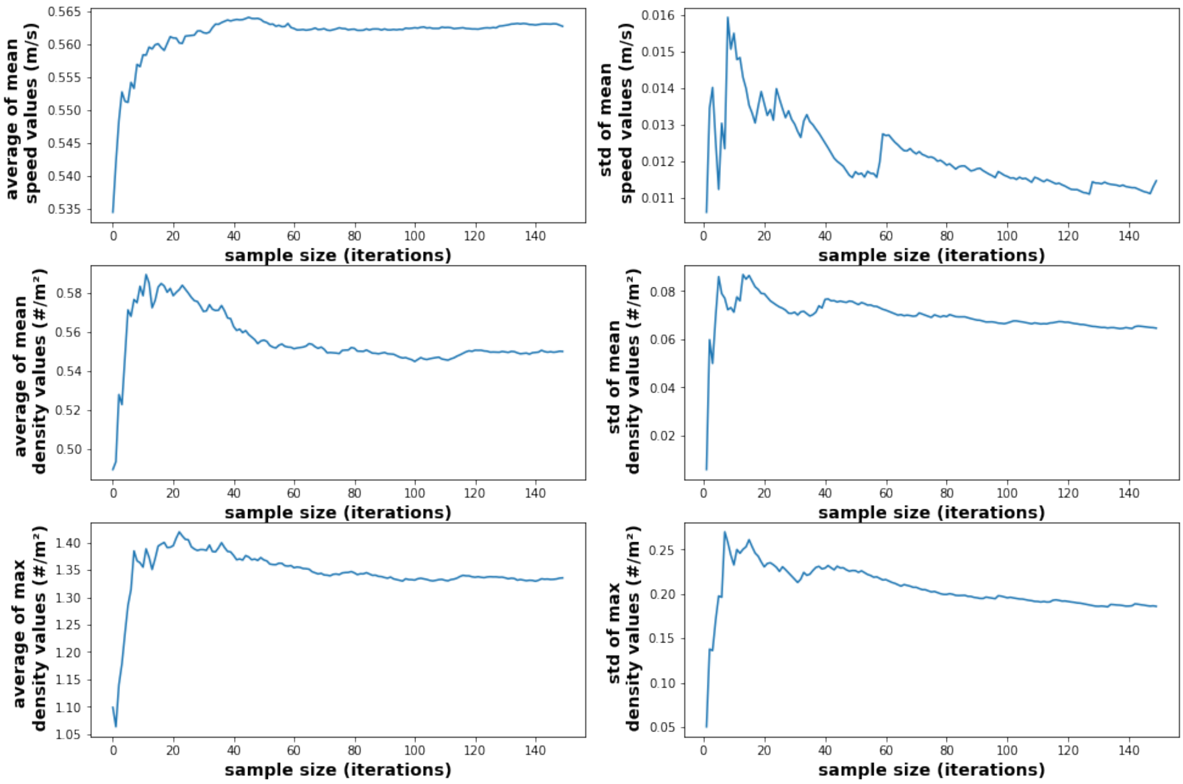


Figure 3.17: Convergence of density and speed in the base case

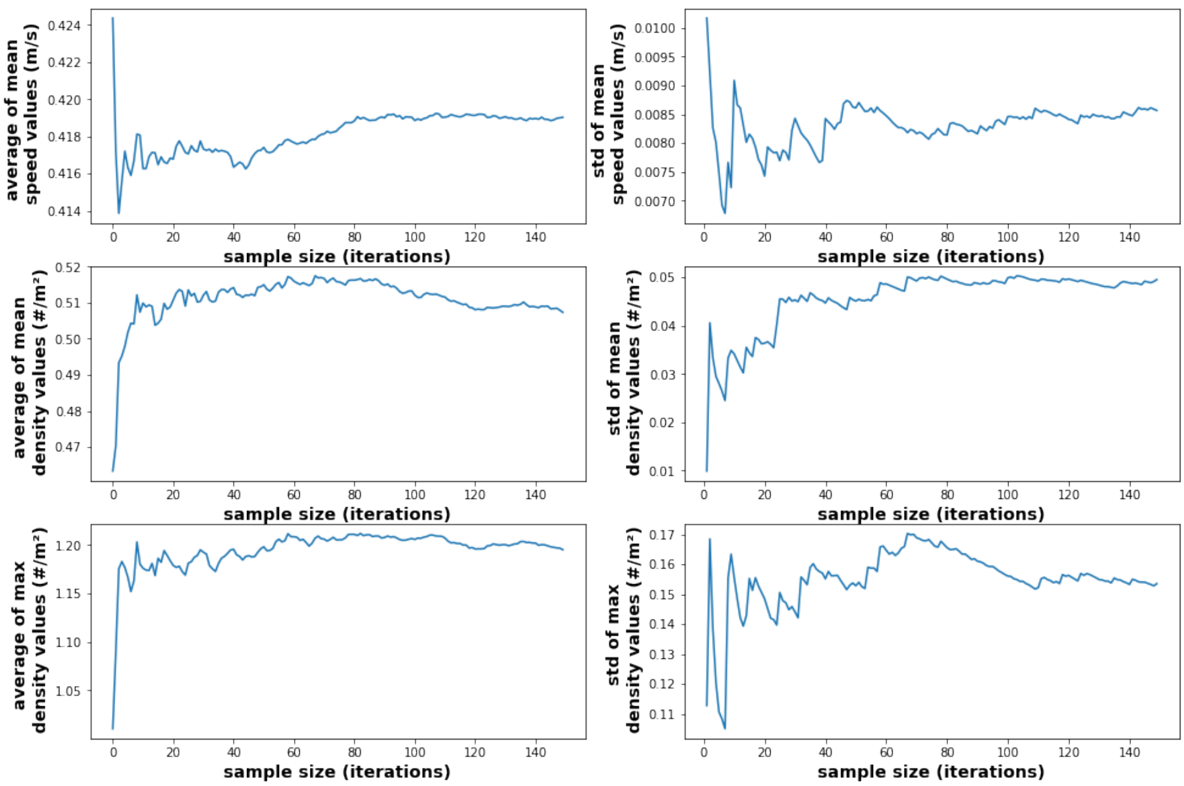


Figure 3.18: Convergence of density and speed in the hypothesized bad case

4

Methodology part 2: Experimental setup

This chapter discusses the experimental setup of this study, and is thereby targeted at answering the third research sub question: *"How can a microscopic crowd model of the Grote Markt, that includes group forming and realistic pedestrian movements, aid operational crowd management?"*. The proposed methodological steps for model experimentation are therefore applied to the case study. This chapter firstly explains the need for the EMA Workbench to realize the application of the proposed steps, with the detailed discussion of the steps itself following afterwards. All created material can be found in more detail on GitHub[1].

4.1. EMA Workbench connector

This section discusses the need for and implementation of the EMA Workbench connector to the Vadere simulator. A short demonstrating of the connector is additionally provided. At the moment of writing, the developed connector and addition to the EMA Workbench source-code, is in the progress of being integrated in the latest release of the EMA Workbench (publically available on GitHub[2]). However, the full source-code is provided on this work's repository hosted on GitHub, including more extensive documentation and a demonstration on the use of the newly constructed model connector.

4.1.1. The need for exploratory modelling tools

In order to move from a crowd model, such as the one constructed for the Grote Markt, to operational support for crowd management, a collection of steps is needed. These steps need to account for the large number of uncertainties and high computational requirements, and enable the generation of useful insights with the model. Three steps are proposed in this work to realize this in practice: exploration, selection, and evaluation. However, to perform these steps, a tool is needed that enabled extensive support on model-based experimentation.

This is where the EMA Workbench, a Python workbench targeted at computational support using models the Exploratory Modelling and Analysis (EMA) Workbench (Kwakkel, 2017), illustrates its potential. The workbench enables quick sampling methods, to deal with large amounts of model uncertainties. Users can thereby perform model experiments directly from Python once the model's input parameters are provided. Moreover, computational requirements can be managed by parallel processing and intelligent sampling methods. In addition, the workbench enables the quick, yet extensive, analysis of model output data by support for extensive analysis tools, such as scenario discovery and sensitivity analysis.

Although this research acknowledges the work previously done on uncertainty quantification with Vadere (Gödel, 2022), the utilization of the EMA Workbench is seen as ideal in this context due to the extensive exploratory modelling support, directly from Python. However, while the EMA Workbench supports connectors to various modelling environments from multiple disciplines, a connection to Vadere is currently not supported out-of-the-box. Therefore, a connection to the EMA Workbench

[1] https://github.com/floristevito/CrowdSim
[2] https://github.com/quaquel/EMAworkbench

is developed, not only to support this work, but also to serve future exploratory modelling approaches using the synergy from the utilization of both Vadere and the EMA Workbench.

4.1.2. Technical implementation

On a high level, the working of an EMA Workbench model connector can be divided into three main steps: (1) initializing or connecting to the model, thereby setting the required paths (e.g., locating of model files) and other needed model specifics; (2) running an experiment with the model, this includes passing a set of parameter values to the model instanced and structuring and storing the return data from the model; (3) cleaning up model output files or other no longer needed temporal data files.

For a functional connector from the EMA Workbench to Vadere, these three roughly outlined steps need to be put into place for Vadere models. Moreover, the mechanisms need to work platform independent, directly from Python. The Vadere simulator relies on Java, but current distribution of Vadere include a console executable (Kleinmeier et al., 2019), enabling to run Vadere from the command line on a machine with a recent Java installation. The build connector subsequently uses this console executable, to perform all three steps. Figure 4.1 gives an overview of the steps performed by the constructed connector for Vadere.

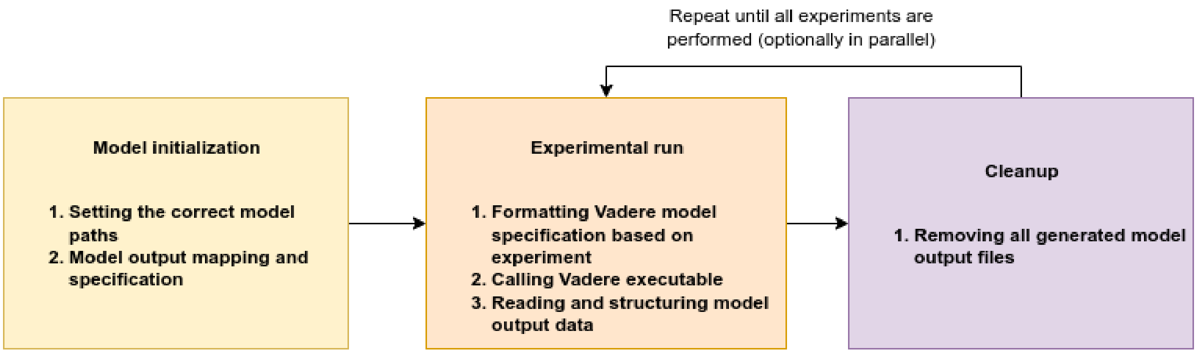


Figure 4.1: Working of the EMA-Vadere connector

First, the connector initializes the Vadere model from Python by setting and storing the correct path names, pointing to the Vadere console executable and model files on the local machine. Additionally, the output files of Vadere need to be specified. In order to collect model output data, Vadere output data processors collect data and store it to specific, user specified, files. The connector thereby assumes that .txt files are used to store scalar outcomes, and .csv are used to refer to time-series outcomes.

The second step is the most complex one in regard to the Vadere connector. In this step, Vadere experiments need to be run, initialized from Python. This is realized by utilization of the Vadere console executable. For each experiment, this executable is called from Python with a specified set of arguments, and a Vadere run is subsequently performed. In order to pass the right parameter values from the sampling, which is performed in Python, special functions are written. These Python functions can map specific parameter values from the EMA workbench data structure to the Vadere model files, which are structured according to the JSON format. In addition, the functions can alter the Vadere model files. These functions thus enable the alteration of Vadere model files, which can subsequently be passed to the Vadere console executable to perform the experimental runs. Finally, this steps reads in the output data files of Vadere, depending on the specified paths from the first step, and structures this data to the format supported by the EMA Workbench.

The final step includes cleaning up model files after each experimental run. In contrast to the more default EMA Workbench approach, where the cleanup is performed after all experiments, the Vadere connector removes all output files after each experimental run. This is implemented in this manner, since the Vadere runs store new output data files for each experiment. To prevent overwriting data and unneeded disk space, the data files are removed after each run.

4.1.3. Usage of the connector

This section covers the basics of using the constructed Vadere model connector, and how a new model with parameters can be set up. Two steps are required for this: (1) initializing the model; and (2) mapping the parameters. In additional to this section, the final model connector, including in-code comments and a Python notebook file with a demonstration on usage, are provided on GitHub and can be consulted for more details.

First, to set up a new Vadere model, the VadereModel class in the EMA Workbench needs to be initialized. This class initialization needs at least the following arguments:

- **name:** A name for the model as a string.
- **vadere_jar:** The path from the wd to the Vadere console executable as a string.
- **processor_files:** A list containing the names of the output data files as strings.
- **model_file:** The path from the wd to the Vadere model file (.scenario) as a string.
- **wd:** The path to the working directory. **Note:** the working directory should not include the Python runfiles, as these will be copied with parallel processing and can take up a significant amount of disk space. Only the vadere executable and model files should preferably be placed in the working directory.

After model initialization, the user can specify which Vadere model variables need to be mapped to the EMA Workbench. In this research, this is done for all nine specified parameters. The connector can subsequently change any variable present in the Vadere model file (.scenario), by specifying the exact location to the variable of interest. Vadere model files are structured similar to JSON objects, and the exact location of a variable can be found by a sequence of nested objects. Thus, for each parameter, the exact location should be passed as a tuple that includes the exact level of nestedness, with the tuple passed as a string. For instance, when mapping “SpawnNumberA” to the corresponding variable in the Vadere model of this work, the argument ’(”scenario”, ”topography”, ”sources”, 0, ”spawnNumber”)’ was provided. This directly points to the spawn number of the first source in the Vadere model, in this case referred to as source A. All model parameters (see Table 3.2) are mapped in this fashion, and a detailed overview of the variable paths are included on the linked GitHub, in the separate EMA formulations Python file.

When the model is fully initialized, and all the parameters are mapped, the connector is ready to use. All sampling methods, experiments with parallel processing, and data-analysis tools can subsequently be performed just as with any model used with the EMA Workbench.

4.2. Application of methodological steps for experimentation

This section discusses the proposed methodological steps for experimentation, utilizing the constructed crowd model of the Grote Markt. The original steps include: exploration, selection and evaluation. This research utilizes techniques supported by the EMA Workbench to apply these steps to the specific case study of the Grote Markt. Four phases that map to the proposed steps are recognized: (1) open exploration; (2) directed search; (3) measurement exploration; and (4) measure evaluation. A detailed overview of the applied steps and related phases is given in Figure 4.2. In addition, more information about the collected model input and output values, together with an overview of the model parameters that are used for experimentation, can be found in Section 3.2.1

Open exploration

First, a phase called open exploration is performed. This entails within the context of this research the exploration of effects of uncertain model input values on the speed and density outcomes, using systematic sampling of the uncertainty space. This step is targeted at informing about what might happen, and what model parameter combinations might lead to certain outcomes. The experimental setup accounts for the uncertainty in the model parameters by using cases sampled over the entire uncertainty space. Table 3.2 gives an overview of the used model parameters, and the ranges in which they are sampled.

However, sampling all possible combinations of input parameters would be unfeasible. Therefore, Latin Hypercube sampling is utilized, with the objective to imitate the uncertainty space with a lower number of samples. While alternative sampling methods could be used, Latin Hypercube sampling is preferred mainly for the method’s simplicity and ease of implementation (Helton & Davis, 2003). Applied

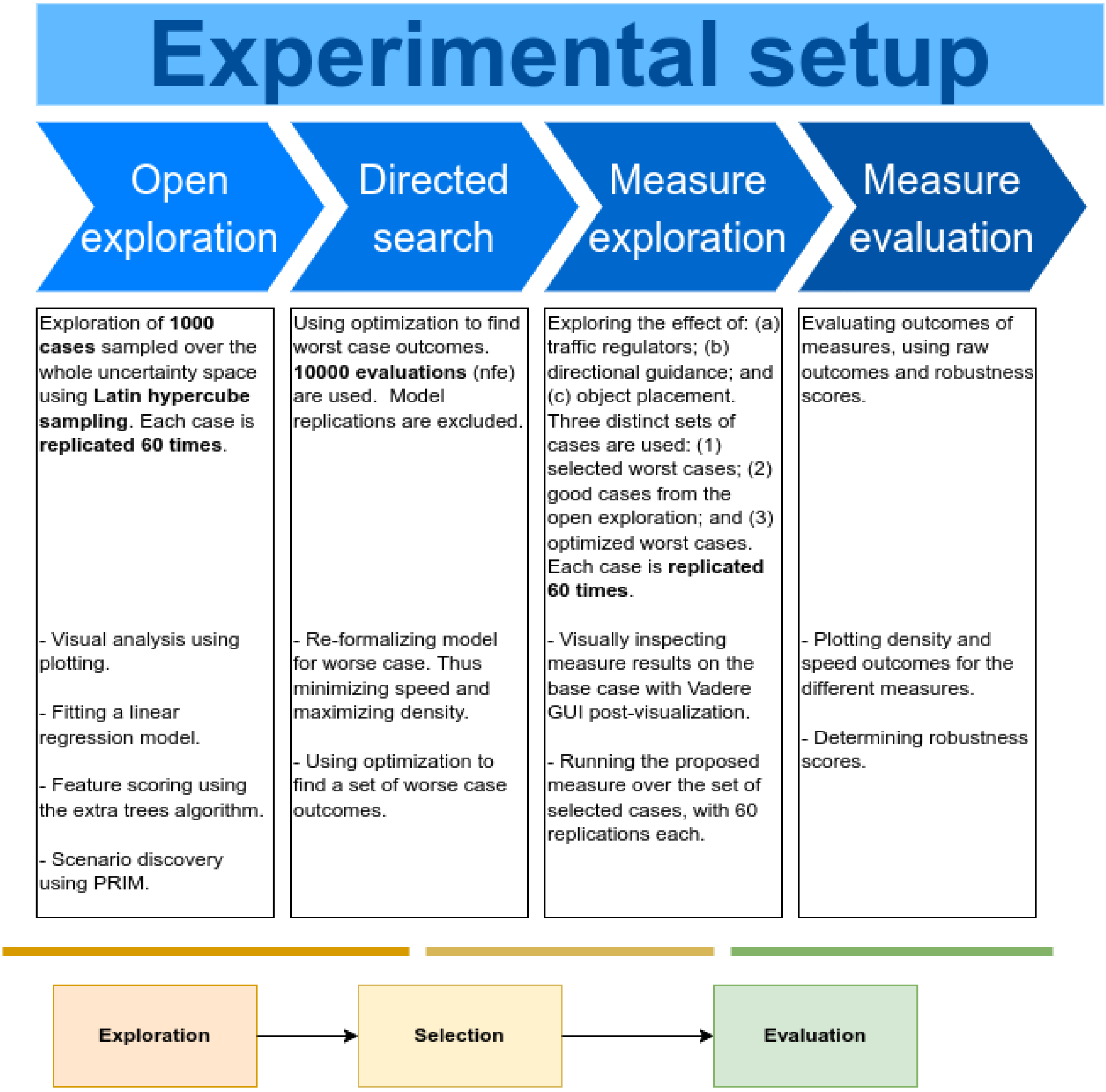


Figure 4.2: Application of the proposed methodological steps for model experimentation

to the case study, 1000 model runs with Latin Hypercube sampling on the uncertainties are performed, each with 60 model replications. To check for representable coverage of the 1000 cases over the uncertainty space, the sampling and evaluation was performed a second time. Analysis indicated similar results, and this research classified the number of 1000 cases accordingly sufficient.

Four analyses are subsequently performed within the open exploration phase. First, visual analyses by plotting the results aid to get a deeper first insight in the mapping between parameters and model outcomes. Second, a linear regression model is fitted, for all the outcomes separately, to explore how much variance in the speed and density outcomes can be explained linearly by the input parameters. Third, feature scoring using the extra trees algorithm (Geurts et al., 2006) is performed to get a better insight in the relevance of the uncertain parameters on the outcomes. Lastly, scenario discovery using PRIM (Kwakkel & Jaxa-Rozen, 2016) is used to discover problematic scenarios. Here, problematic cases are based on the density threshold of more than one pedestrian per square meter (Schaap & Dijkshoorn, 2020, p. 15). Problematic cases in this research are classified as those where the maximum density in one of the four measurement areas succeeds one pedestrian per square during the experimental runtime (100 seconds).

Directed search

As the second experimental step, a directed search phase is performed. This phase within this research includes the use of optimization functions to find problematic cases. As seen in the work of Halim et al. (2016), an optimization over the uncertainties is performed, to find those cases that lead to the worst model outcome values. In contrast to the open exploration, where these cases are sampled systematically, optimization is now used to search for cases where the maximum densities are high, and the walking speed is low. Since a directed search using optimization is heavily computationally demanding, model replications are excluded in this step. Besides, results from the found set of cases are re-evaluated with in-event crowd management measures in a later stage, including replications.

Within this research, to find problematic cases, the pedestrian speed is minimized, while the maximum density values are maximized. The number of evaluations, known as NFE, is changed subsequently to a number of 1000 evaluations, where convergence of results was reached. Moreover, small epsilon values are used, 0.2 for all model outcomes.

Measure exploration

The third experimental step consists of the exploration of in-event crowd management measures. Three sets of cases are hereby utilized. First, identified problematic cases from the open exploration phase, with maximum density values that succeed the one pedestrian per square meter threshold, are used as a first set. Second, identified good cases from the open exploration phase are used as a control set. This set is relevant in order to check for possible adverse effects of in-event crowd management measures in goods cases. Furthermore, the optimized, extreme, worst cases are includes as a third set. Lastly, in regard to computational limits, the first and second set of cases, sampled from the open exploration phase, are limited to 100 cases per set respectively.

A larger set of cases from the problematic and good case categories would lead to significant computational restraints, and would additionally not be needed for a valid assessment of the in-event crowd management measures under different circumstances. For this reason, a random sample of cases is gathered over the classified good or bad cases. While this could potentially lead to limited coverage of the whole uncertainty space, this work trusts in the representativeness of the sets of cases, due to the high diversity and include optimized cases. Furthermore, the random sampling of good and bad cases is repeated in isolation, and found variance in aggregated outcomes was so low that the set of used cases are classified as representable.

The distinction in the set of cases between unsafe problematic cases and good reference cases serves an important purpose in this research. In the real-world use cases, the proposed crowd management measures are aimed to be in-event. That is, they would be utilized only in cases when they are necessary. Existing crowd management platforms, using monitoring techniques, could for example identify when they would be needed. Hence, in-event crowd management measures are aimed to improve problematic cases. Whether they improve cases that are not seen as problematic is seen as less relevant in this regard. However, to analyze whether in-event crowd management measures could potentially have negative effects under good conditions, a set of good reference cases is also used. Overall, the total set of sampled cases over the uncertainty space is evaluated for each crowd management measure. Model results are again averaged over 60 replications for each case.

Measure evaluation

The last experimental step consists of the evaluation of the in-event crowd management measures. Resulting density and speed values of the in-event crowd management measures are evaluated and compared to each other and to cases where no crowd management measure is utilized (zero case). Additionally, robustness of the in-event crowd management measures over the set of problematic cases is considered. This is done by the utilization of two different robustness metrics. First, problematic cases consist of those cases where the maximum density in one of the four measurement areas succeeds one pedestrian per square meter. The first kind of robustness metric assesses whether this violation of density in each of the four density areas is averted completely, and sums the number of areas that meet this density criteria. This leads to a number between zero and four for each tested case. Hence, the first robustness metrics indicates the absolute number of areas with a mean density classified as safe over the total set of cases tested, in addition to the count of completely averted cases (with no area that reaches a density higher than one pedestrian per square meter). However, this metric does not indicate the potential magnitude of density and additional speed improvements. Therefore, a second kind of robustness metric is used. This second metric is referred to in this work as the mean-variance metric. Based on the calculation used in the work of (Hamarat et al., 2014, p. 33), this metric is targeted at informing about the potential magnitude of a model outcome, adjusted by the variance, and calculated as follows: dividing the mean by the standard deviation for maximization, and multiplying the mean by the standard deviation for minimization. This additional robustness metric is calculated for the mean speed value, along with both the maximum density values.

5
Results

This chapter presents model results acquired by the application of the proposed methodological steps on the case of the Grote Markt. First, results from the open exploration phase of the model, without in-event crowd management measures, are presented. This includes a general discovery of results as well as an extensive scenario discovery. Three distinct sets of cases are used to explore the effect of the three proposed in-event crowd management measures at the Grote Markt: (1) traffic regulators; (2) directional guidance; and (3) object placement. Together, the presented results answer the last sub question of this research: *"How do the in-event crowd management measures at the Grote Markt perform, given the inherent uncertainty of crowd modelling?"*.

5.1. Exploration: The Grote Markt without crowd management

This section presents model results of the Grote Markt, acquired without any of the proposed in-event crowd management measures, and with model parameters sampled over the entire uncertainty space. First, an extensive insight into the effect of model parameters on density and speed outcomes is given. Afterwards, distinct sets of cases are selected, on which the proposed in-event crowd management measures are evaluated afterwards.

5.1.1. Open exploration

As illustrated in Table 3.2, a large set of model parameters, each with a wide range of possible values, are included in the open exploration of the model. Four techniques are used to get an insight into the effect of these parameters on modelled density and speed values: (1) visual plotting; (2) regression models; (3) feature scoring; and (4) PRIM based scenario discovery.

First, visually plotting model outcomes against each other enabled an initial insight into the all possible combinations of model outcomes, the average speed and the eight density-based outcomes, and potential correlations between them. Overall, strong correlations were found between maximum and mean density outcomes. Model runs that lead to higher maximum density values, generally also found a higher density on average over the whole run. Figure 5.1 illustrates one of the found correlations, with the found maximum density values scattered against the mean densities found in measurement area 1. The maximum and mean outcomes show a strong correlation. However, interesting results can be found when inspecting the mean speed values, colored from dark to light. Higher speed values seem to cluster in the middle, while one would expect optimal speed values in cases where densities are low and hence more free space is available. Note that this finding is likely due to the lack of model warm-up time, increasing the average values found, as discussed in more detail in Section 6.1.1.

Second, fitted regression models, with the speed and densities as dependent variables, and all nine model parameters as independent variables, generally resulted in a high r-squared value. Especially the model fitted on the mean speed values with an r-squared of 0.92. Hence, a high amount of variance in the speed and density values can be explained linearly by the nine model parameters. Additionally, it was found that the group forming generally has a negative effect on the speed outcomes, while positively influencing the measured densities.

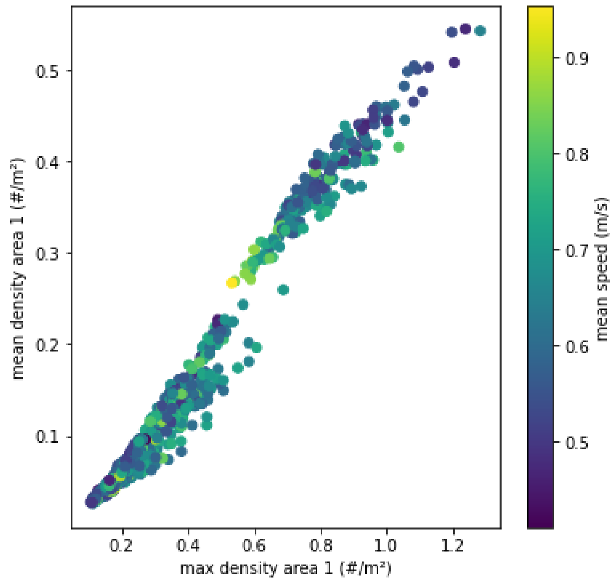


Figure 5.1: Scatter plot of mean and maximum densities found in area 1

Note. Besides the strong correlation between the density outcomes, this plot also indicates a surpising clustering of mean speed values around higher density outcomes. This latter effect is likely due to model specific implementation, as discussed in Section 6.1.1

To enable a better comparative analysis between the effects of input parameters, feature scoring is performed next. Figure 5.2 illustrates the results of feature scoring using the raw outcome values. Additionally, Table 5.1 illustrates feature scoring performed with binary classification. Here, model outcomes where one of the four density areas succeed one pedestrian per square meter were classified as problematic. Subsequently, feature scoring on these binary outcomes resulted in a slightly different outcome. Both analyses illustrate the importance of the spawn frequency (controlling the number of pedestrians generated each second) on the found density and speed outcomes. However, feature scoring using binary classification emphasized the importance of the spawn frequency specifically for sources D, C, and A.

Table 5.1: Feature scores using binary classification of the results

Parameter	Feature score
spawnFrequencyD	0.33
spawnFrequencyC	0.32
spawnFrequencyA	0.13
obstPotentialHeight	0.04
pedPotentialHeight	0.04
groupForming	0.04
sdFreeFlowSpeed	0.04
meanFreeFlowSpeed	0.04
spawnFrequencyB	0.03

Lastly, model outcomes are analyzed using the PRIM algorithm (Kwakkel & Jaxa-Rozen, 2016). This algorithm used model outcomes that were classified according to the safety threshold in this research, where one of the four density measurement areas resulted in a density that exceeds one pedestrian per square meter. Using PRIM, a box can be discovered that makes a tradeoff between coverage and density. Coverage hereby illustrates the number of values of interest that are included in this box. Density denotes the share of values of interest over the total set of values in the box.

Figure 5.3 illustrates the three boxes subsequently found within this research. All with a density of 1, meaning that, by restricting the respective dimensions, these boxes can explain a portion of the

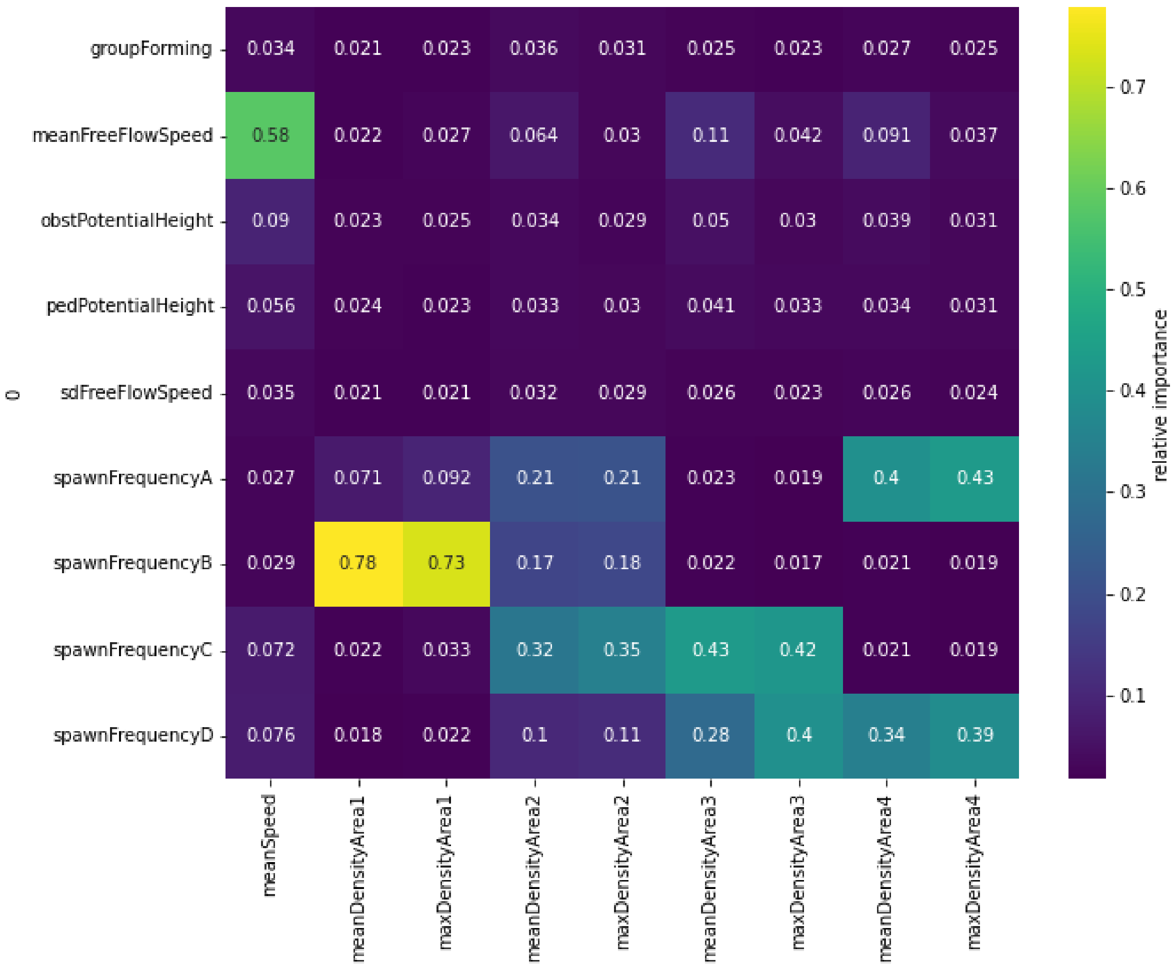


Figure 5.2: Feature scores using raw outcome values

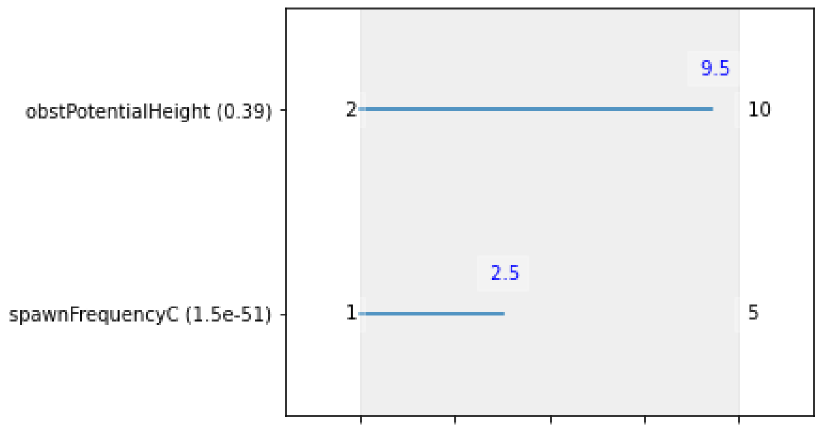
Note. Values represent relative feature importance and are color coded accordingly.

cases of interest without including any cases that are not of interest. The first box covers roughly 52% of the cases of interest. The second box described about 30% of the remainder of the cases, and the third box almost 10% of the second set of residual cases. For the remaining cases, a fourth box with a density above 0.8 could not be found. Considering the given coverages of the first three boxes, only 56 of the initial 725 cases of interest are not yet explained by one of the three boxes.

Besides the amount of generated pedestrians, PRIM points to the importance of the personal space and the object repulsion factor of pedestrians. However, while the restricted dimensions concerning the number of pedestrians within this box are found to be statistically significant by proxy (see the quasi p-values, illustrated next to the variable names), this was not the case for the strived personal space and space from obstacles. Meaning that, based on these results, the respective restricted dimensions in isolation cannot be seen as significant. These findings are, however, in line with the found feature scores using binary classification (see Table 5.1), where mainly the spawn frequency D, C, and A are of importance, with less relevance to the remaining features.

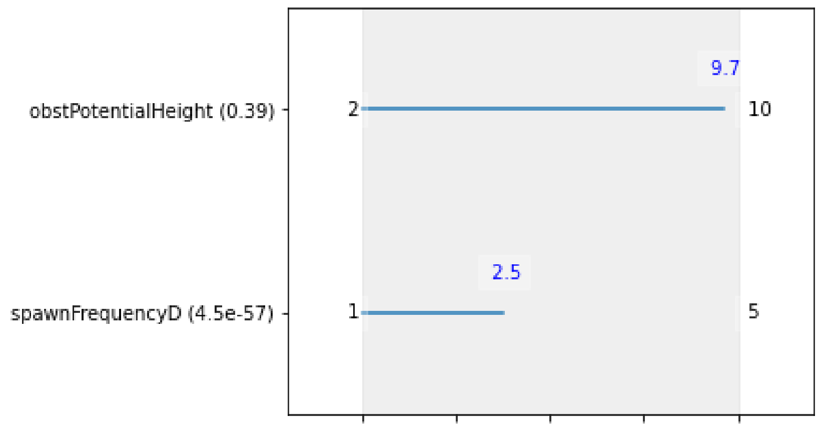
5.1.2. Directed search

In addition to the systematic sampling in the open exploration, a directed search ensured the inclusion of extreme worst cases. This resulted subsequently in a Pareto optimal set of 10 most problematic cases on either the speed outcome or one of the four maximum densities. Figure 5.4 illustrates these cases, with their score on the nine model outcomes. Often, these cases show a high number of pedestrians and a highly desired distance between others (personal space) and objects (object repulsion).



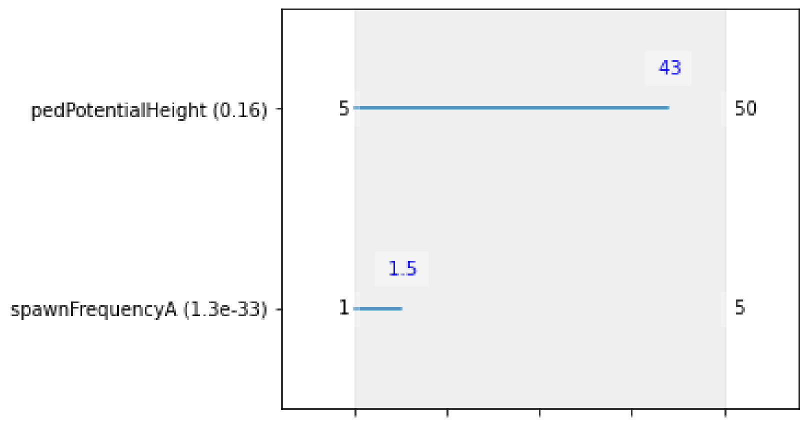
(a) PRIM box 1: Coverage 0.52 & density 1

*covering 52 % of the cases (**coverage**), with 100% of the included cases of interest (**density**)*



(b) PRIM box 2: Coverage 0.30 & density 1

covering 30% of the remaining cases, with 100% of the included cases of interest.



(c) PRIM box 3: Coverage 0.09 & density 1

covering roughly 10% of the residual set of cases, with 100% of the included cases of interest.

Figure 5.3: Scenario discovery with PRIM

Note. Box figures denoting: (1) the restricted dimension on the y-axis, accompinied by their quasi p-value; (2) the entire uncertainty space in gray; and (3) the bandwidth of the restricted dimension by a blue line.

However, resulting worst cases often also indicated a relatively high number of group forming. In seven out of the ten worst cases, group forming was found to be higher than 70%.

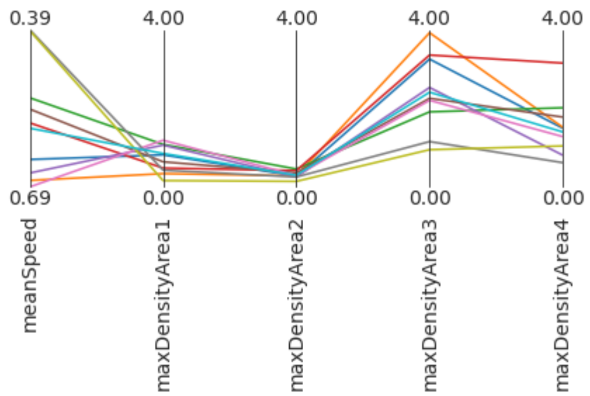


Figure 5.4: Parallel coordinate plot of outcomes from the optimized worst cases

Note. Axis are inverted to ensure that undesireable, worst case, outcomes are portayed in the same direction.

5.2. Selection: Problematic, good, and worst cases

Three distinct sets of cases are selected to evaluate the in-event crowd management measures on resulting density and speed values: (1) identified problematic cases from the open exploration phase, with maximum density values that succeed the one pedestrian per square meter threshold; (2) identified "good" cases from the open exploration phase, represented by cases that did not exceed the density threshold; and (3) the optimized extreme worst cases, those cases found by the directed search that result in the highest density or speed outcomes. Note that the first and second sets of cases are sampled from the initial open exploration sample of thousand LHS cases over the whole uncertainty space. These cases are then resampled randomly, to reduce the number to 100 cases per set.

5.3. Evaluation: In-event crowd management measures

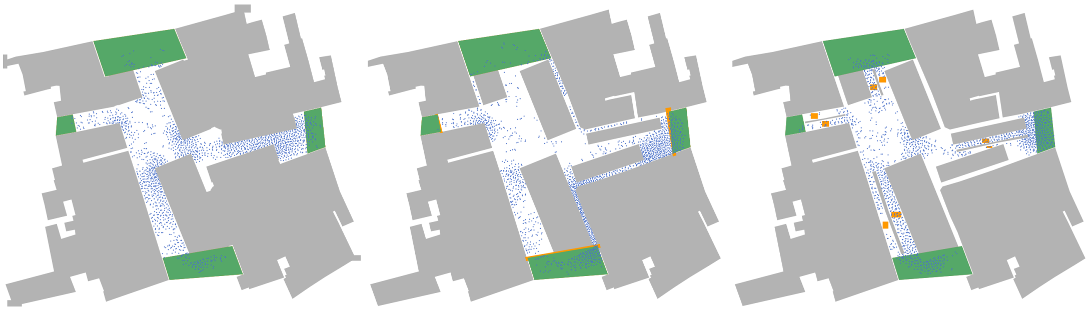
The open exploration phase enabled a better insight into the effect of model parameters on the found density and speed outcomes, without any in-event crowd management measures. This section firstly illustrates the effect of the three implemented in-event crowd management measures: (1) traffic regulators; (2) directional guidance; and (3) object placement. This is done visually using the Vadere GUI and post-visualization tool for the base case first. Subsequently, raw outcomes and robustness scores are discussed.

5.3.1. Effectiveness of crowd management

Figure 5.5 illustrates the effect of the three in-event crowd management measures visually on pedestrians at the Grote Markt, following the base case (see Table 3.2). It can be observed visually that the three in-event crowd management measures have highly diverging effects on the spatial distribution of pedestrians after 90 seconds in the simulation. High density clusters still occur, but at different spatial locations depending on the utilized crowd management measure. New, seemingly high density areas can be observed in the directional guidance (Figure 5.5b) and object placement (Figure 5.5c) measures compared to initial runs (see Figure 3.8). The traffic regulator measure (Figure 5.5a) initially seems most optimal in the base case, distributing pedestrians more evenly over the available walking space.

The effects of the in-event crowd management measures on the raw mean speed outcomes are illustrated in Figure 5.6. Three main points can be observed. First, most of the in-event crowd management measures do not improve speed outcomes compared to the zero case. Second, the decrease in speed caused by the measured, compared to the zero case, is higher under problematic and worst cases, and less so under good cases. Third, the regulators measure does improve speed under all three sets of cases, in contrast to the other measure. Under all sets of cases, this measure improved the average speed by roughly 0.05 m/s.

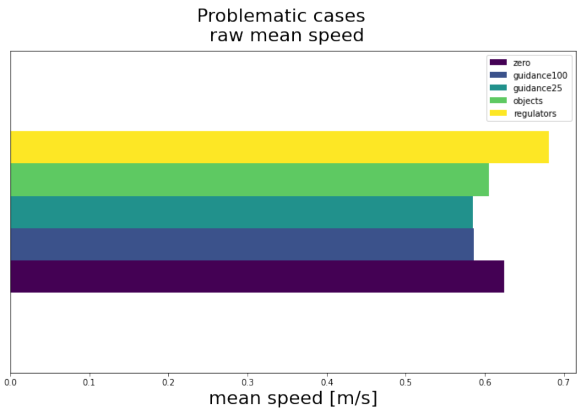
Figure 5.7 illustrates the modelled raw maximum density values over the different sets of cases. It is important to note in this regard that lower values are conceived as better, since this means that



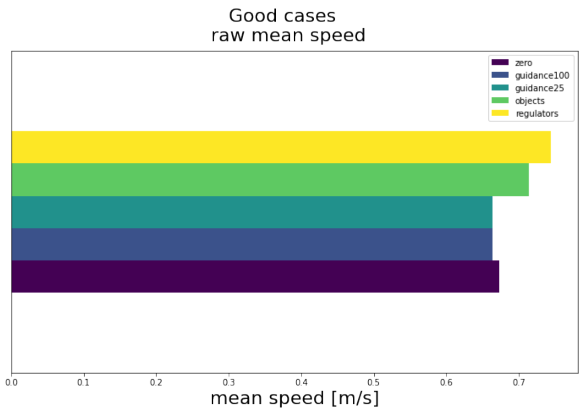
(a) Traffic regulators (b) Directional guidance (c) Object placement

Figure 5.5: Crowd management measures visualized after 90 seconds in the base case

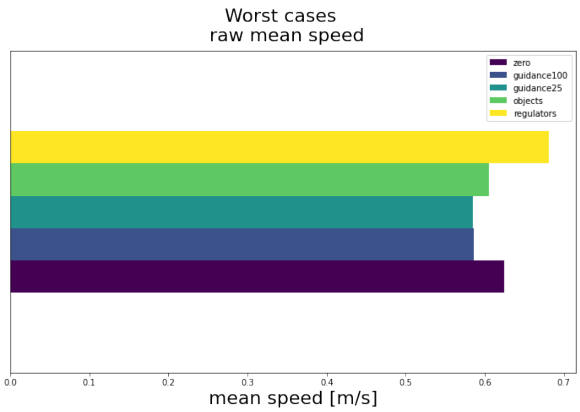
(a) Traffic regulators, closing the entrances of narrow spaces between terraces. (b) Directional guidance, changing the destination of pedestrians. (c) Object placement, line barriers in multi directional corridors.



(a) Problematic cases



(b) Good cases

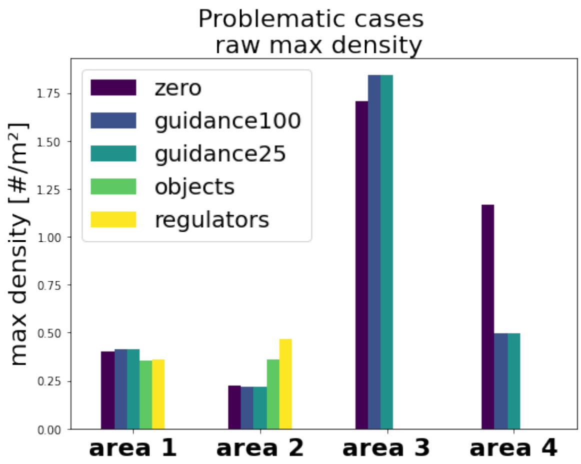


(c) worst cases

Figure 5.6: Effect of crowd management measures on raw mean speed

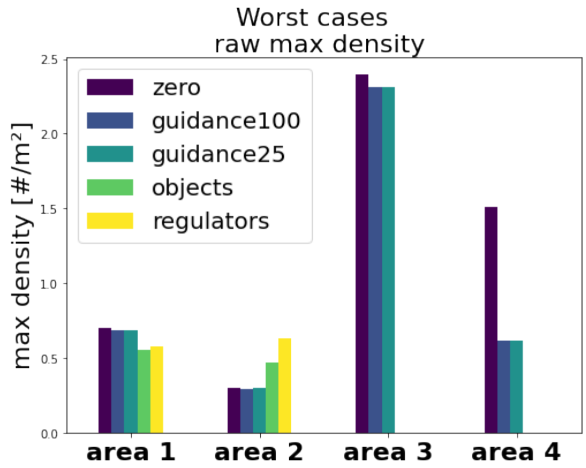
(a) Average performance over sample of 100 problematic cases from open exploration phase. (b) Average performance over sample of 100 good cases from the open exploration phase. (c) Average performance over 10 optimized extreme worst cases.

maximum density values found over the runtime are lower. Results point to five main observations. First, under problematic cases, the regulator and object measure lower the density found in the third and fourth areas to zero. Second, the regulators and object measure do lead to increments of the densities found in the second area, but not in the first area. This observation is important, since both measure close problematic areas (3 and 4), making it thereby crucial to check density increments in the other areas. Third, overall similar behavior can be observed under the problematic and good cases, but not under worst cases. This specifically applies to route guidance, which only really lowers the density in the fourth measurement area, and even increases the values in the third measurement area under the problematic and good cases. Yet, under extreme worst case circumstances, this effect seems to shift, where route guidance decreases densities found in both the third and fourth measurement areas.



(a) Problematic cases

(b) Good cases



(c) worst cases

Figure 5.7: Effect of crowd management measures on raw maximum density values

(a) Average performance over sample of 100 problematic cases from open exploration phase. (b) Average performance over sample of 100 good cases from the open exploration phase. (c) Average performance over 10 optimized extreme worst cases.

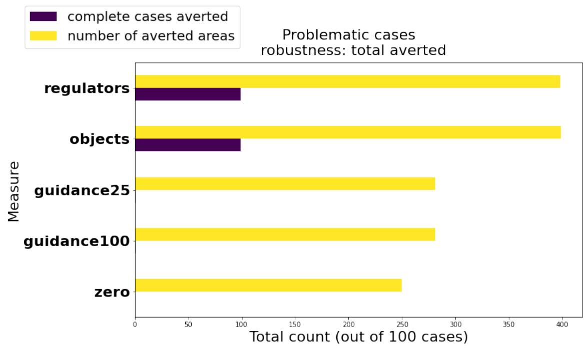
5.3.2. Robustness of in-event crowd management measures

To get a deeper insight into the effects of the in-event crowd management measures, and their robustness of working under diverse circumstances, this research proposed two kinds of robustness metrics. First, the number of averted cases portrays the number of absolute cases which are classified as safe. Meaning that these cases do not reach the threshold density values in all four measurement areas. Additionally, this metric also includes the sum of measurement areas that are now classified as safe (areas averted). Second, the mean variance score illustrates the mean outcome adjusted for the variation in outcomes.

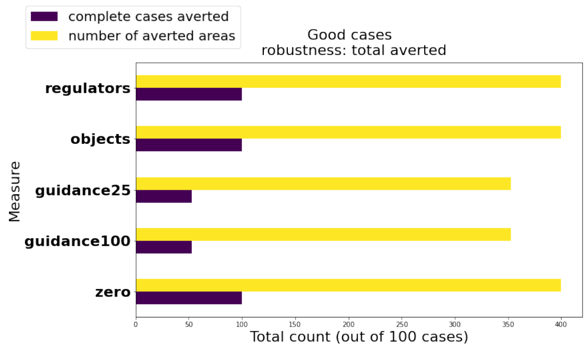
Averted cases

The number of averted cases, in terms of preventing the threshold density value, is illustrated in Figure 5.8. This figure refers to two things: (1) the complete cases averted, entailing cases that did not exceed the density threshold in all four measurement areas; and (2) the number of averted areas, representing the total count of measurement areas that did not exceed the density threshold.

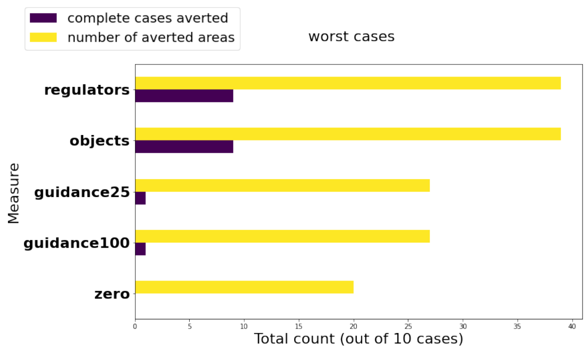
Four remarks can be made based on the results presented in Figure 5.8. First, the zero case shows no aversion under both problematic and worst cases, with a complete aversion score of 100 under good cases and 0 under problematic cases (n=100). Second, the regulator and object measures score best, with nearly 100% aversion under all three sets of cases. Third, the route guidance measure scores relatively poorly compared to the other measures, but almost always slightly better than the zero case. Fourth, route guidance does decrease performance under good cases. Not only is this visible by the lower number of totally averted areas, a lower number of completely averted cases is in addition found, compared to the zero case.



(a) Problematic cases



(b) Good cases



(c) worst cases

Figure 5.8: Robustness scores: Averted problematic cases

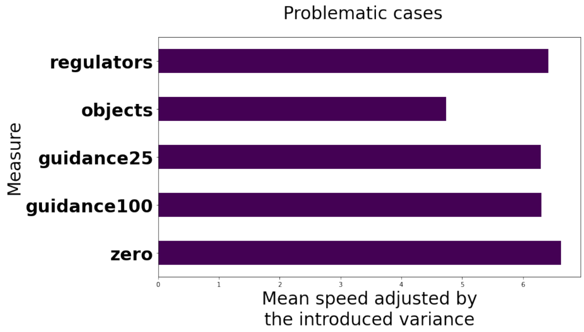
(a) Performance over sample of 100 problematic cases from open exploration phase. (b) Performance over sample of 100 good cases from the open exploration phase. (c) Performance over 10 optimized extreme worst cases.

Mean variance scores

The mean variance scores are evaluated separately for both the speed and density values. Whereas speed values are desired to be higher, density values should be kept low. Therefore, the mean variance score for speed is acquired by dividing the mean score by the standard deviation. The score for the density values is, in contrast, calculated by multiplying the mean value by the standard deviation. Hence, regarding the mean variance scores for speed, higher values are desirable. Contrastingly, for density, lower mean variance scores are desired.

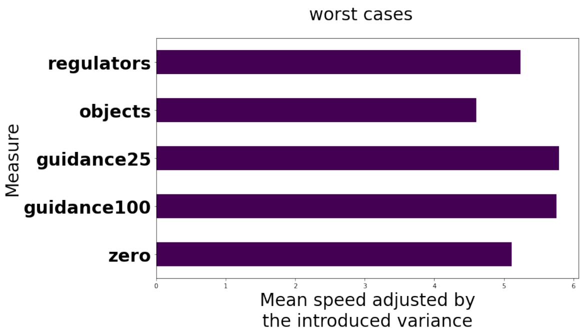
Resulting speed values, illustrated by Figure 5.9, are similar to the raw speed outcomes that were not adjusted by introduced variance (see Figure 5.6), with the exception of two aspects. First, where the regulator measure improved the speed value under all cases without adjusting for the introduced variance, the measure was found to decrease speed under problematic cases when adjusted for variance. Second, the route guidance measure performs the best of all measure under worst cases, while it was found to be the worst scoring measuring on the raw outcomes.

The maximum density values, illustrated by Figure 5.10, indicate similar behavior of the testes in-event crowd management measures without adjusting for the introduced variance (see Figure 5.7), with one exception. The route guidance measure, adjusted by the introduced variance, improved the density found in the third measurement area under problematic cases. In contrast to the raw outcomes, where the route guidance increased the density compared to the zero case.



(a) Problematic cases

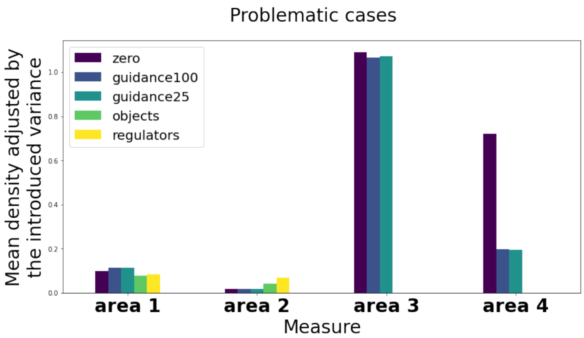
(b) Good cases



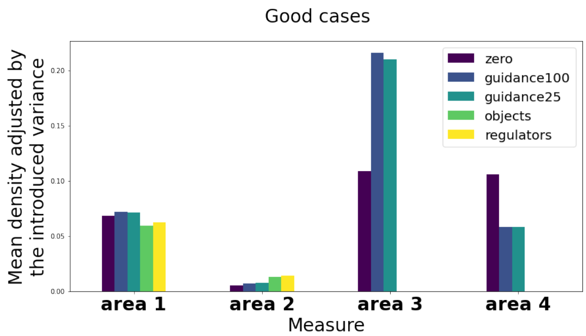
(c) worst cases

Figure 5.9: Robustness scores: Mean variance score for speed outcomes

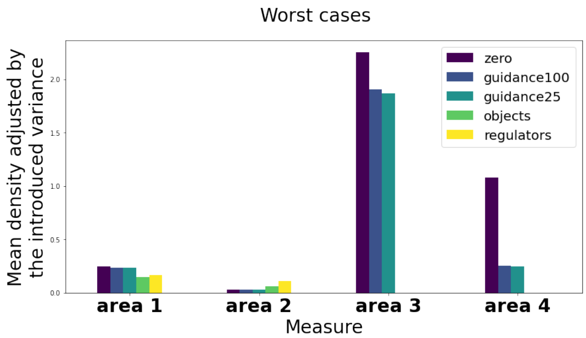
(a) Performance over sample of 100 problematic cases from open exploration phase. (b) Performance over sample of 100 good cases from the open exploration phase. (c) Performance over 10 optimized extreme worst cases.



(a) Problematic cases



(b) Good cases



(c) worst cases

Figure 5.10: Robustness scores: Mean variance score for density outcomes

(a) Performance over sample of 100 problematic cases from open exploration phase. (b) Performance over sample of 100 good cases from the open exploration phase. (c) Performance over 10 optimized extreme worst cases.

6
Conclusion & discussion

Operational support for in-event crowd management is sparse (Wijermans et al., 2016). Models can help understand in-event crowd management system more thoroughly (Sharma et al., 2018), and illustrate potential effects of measures before they have to be implemented in real life. However, crowd models are complex to fully utilize for operational insights. This research proposed three methodological steps—exploration, selection, and evaluation—for model-based experimentation to overcome current challenges in the application of crowd models. These steps were then applied to a case study of operational crowd management at the Grote Markt, in the city of Breda. The research question thereby was: *"What effect do the in-event crowd management measures—traffic regulators, directional guidance, and object placement—have on the density and walking speed of pedestrians in the Grote Markt, Breda?"*. To answer this question, this study: (1) constructed a detailed microscopic crowd model of the Grote Markt; and (2) applied this model according to the proposed steps for model-based experimentation. This chapter discusses the results found, followed by the strengths, limitations, points for further research, and implications of this work. Finally, a conclusion of this work is presented.

6.1. Discussion

Four sub questions were formulated to analyze the effects of in-event crowd management at the Grote Markt. First, the roles and limitations of crowd models for in-event crowd management were analyzed. This resulted in the proposal of three methodological steps—exploration, selection, and evaluation—for experimentation, targeted at utilizing crowd models for in-event crowd management. Second, a model was constructed for the case study of the Grote Markt. Third, an experimental setup for this model was specified based on the proposed methodological steps for experimentation. Fourth, the performance of the three in-event crowd management measures was analyzed using the constructed model and proposed steps for experimentation. This section discusses the research results, structured on the three proposed methodological steps for experimentation: (1) exploration; (2) selection; and (3) evaluation.

6.1.1. Exploration: Density and speed without crowd management

The first phase consisted of the exploration of the crowd model without crowd management in place. Four analyses were performed with experimental runs using systematic (LHS) sampling over the entire uncertainty space: (1) visual analysis; (2) linear regression analysis; (3) feature scoring; and (4) scenario discovery using PRIM. Lastly, a directed search by optimizing over the model uncertainties (parameters), targeted at finding worst cases, was performed.

The first analysis, visual plotting model outcomes, enabled a rapid insight into the relationships between model outcomes and the clustering of results. Mean and maximum density values indicated strong correlations, as expected, since these cases are likely subject to higher amounts of pedestrians. Results, however, illustrated a positive correlation of the mean speed and mean density values, which seems invalid given known insights on the relation between pedestrian flows and densities, even in bidirectional shopping streets (Zhang et al., 2011).

In this work, the positive effect of the free flow speed on the mean densities found is likely due to model specific implementation. Since it is complex to determine an initial state of the model, one where pedestrians are already present on natural places in the shopping street, the model starts with an empty model area. This setup would argue for the use of a warm-up time (Grassmann, 2009), where model statistics are gathered only after a certain amount of time. However, a warm-up time was currently not possible to implement within the scope of this work. As a direct result, with higher speed values, pedestrians reach areas where bottlenecks occur earlier, and thereby inflate the mean density values found over the total simulation time.

The next three analyses provided a more in-depth understanding of the relation between model uncertainties (or parameters) and the measured speed and density. Linear regression models and feature scores calculated with the Extra Trees Algorithm indicated the relative importance of model parameters. In addition, scenario discovery using PRIM illustrated exact ranges of model parameters that covered a specific subset of problematic model outcomes, where the density exceeded the 1 pedestrian per square meter threshold. Three overall points of discussion are highlighted based on these analyses.

First, results conformed the positive relation between the mean speed model parameter and the measured mean density. It was found that the maximum densities were not impacted as much by this model specific implementation. Therefore, when discussing density, this research focused on the measured maximum density values.

Second, this work observed by far the highest influence on the measured densities to be the number of pedestrians origination from one of the four connecting streets of the Grote Markt. While the number of agents proved to be non-influential in the work of Gödel (2022) on flow speeds in a bottleneck, its influence on the density on the shopping streets is recognized in related works (Echeverría-Huarte et al., 2021). Specific to the Grote Markt, this research highlighted the importance of the number of pedestrians originating from the Veemarktstraat or the south side of the Grote Markt area. While these findings are prone to the spatial location of the density measurement areas, the found effect was so strong that these origins of visitors could be a fruitful starting point to indicate potential unsafe situations.

Third, results pointed to the main importance of the free flow speed and distance kept from obstacles and other pedestrians (personal space) on the resulting average speed. While the high free flow speed is logically directly related to the concluding speed, the strong relation between the distance kept between obstacles and other pedestrians on the walking speed is one found in more recent crowd studies (Echeverría-Huarte et al., 2021; Gödel, 2022; Mayr & Köster, 2021), and can be a crucial insight for decision makers. When pedestrians could be conveyed to pursue personal space around them less strongly, this could impact the flow on the shopping streets. Furthermore, when certain developments such as the one and a half meter rule during the COVID-19 crisis (Groeneweg, 2021) present themselves, more issues with the speed of pedestrians on shopping streets can be expected.

Regarding the fourth analysis, a directed search over the model uncertainties (parameters) resulted in a set of ten worst cases. These cases are indicated either by a high maximum density in at least one of the four measurement areas, or by a low overall walking speed. Many of the resulting cases were found with a high number of pedestrians and/or a highly desired distance between others (personal space) and objects (object repulsion). This is in line with the results from the previous analysis.

Furthermore, the directed search pointed to the importance of group forming in the model, since most of the found worst case scenarios included a large amount of group forming. While previous analysis illustrated low importance of group forming, this is likely due to the overshadowing by the number of pedestrians in the simulation, which proved to be the most crucial. However, group forming is an important element to consider regarding the valid modelling of crowds (Köster et al., 2011; Lu et al., 2017).

Lastly, it is important to note that both the distance kept between obstacles and other pedestrians are included in the model by the strength of potential functions. That is, when a pedestrian finds its next optimal step in the model, the relative importance of the closeness to other pedestrians or buildings is changed. This strength factor thus does not represent a specific and constant distance. More concretely, this means that, especially in regard to the strong effect of the distance from buildings, the narrow passages between the buildings and terraces on the Grote Markt possibly overestimate the established relation. For instance, when pedestrians assign a unrealistically high importance to the distance from buildings, they could get stuck in extremely narrow passages. Especially when, in addition, a high importance is given to distance from other pedestrians.

6.1.2. Selection: Problematic, good, and worst cases

This work selected three distinct sets of cases: (1) good cases, based on those cases from the open exploration that did not exceed the density safety threshold; (2) problematic cases, based the open exploration cases that did exceed the density threshold; and (3); worst cases, represented by the ten optimized worst cases from the directed search.

An important element of this case selection phase was the possible reduction of cases to evaluate in-event crowd management measures on, while still covering the entire uncertainty space. This work thereby reduced the total set of 1000 cases sampled over the entire uncertainty space to a set of 210 cases. Of this number, 10 cases are included for the worst cases. The remaining 200 cases are equally divided over the good and problematic cases, by randomly sampling these sets after classification. The representativeness of these sets was confirmed by repeating the random sampling in isolation, with analyses illustrating similar results.

While the number of cases was reduced significantly, representativity of those cases should always be guarded. This work has confidence in the reduced sets of cases, but recognizes the possible exclusionary of certain cases. Because this work used a broad range of model parameters, and subsequently focused on problematic density outcomes, it is expected that most cases represent a rather large number of pedestrians. The set of good cases is therefore included to represent circumstance with less problematic elements, such as with a lower number of pedestrians.

6.1.3. Evaluation: In-event crowd management at the Grote Markt

Three in-event crowd management measures, aimed at application at the Grote Markt, are proposed in this research. First, (1) traffic regulators are proposed, who have the ability to block strategic parts of the walking space for a subset of pedestrians. Second, (2) directional guidance is implemented, based on changing the destinations of pedestrians. Finally, (3) the object placement measure placed line-barrier shaped objects within the pedestrian walking spaces.

It was found that, overall, the traffic regulators outperformed the other measures. Not only did this measure improve speed and density values under problematic, good, and worst cases, the measure also mitigated many situations classified as unsafe (without crowd management) to safe circumstances. By enforcing this measure, problematic density areas could be closed off without increasing density in the other areas, and without reducing the overall speed. However, it is important to note that, while not included in the currently used model, the blockage by traffic regulators could break when a crowd potentially gets angry or too large (Park et al., 2018). This would mean that pedestrians can pass through eventually. In addition, visualizations of the clustering of pedestrians pointed to the possibility that new problematic density areas form at other locations.

The other two proposed in-event crowd management measures showed less potential compared to the traffic regulators. Similar behavior was initially found for the object placement measure, performing overall good on reducing the maximum densities, especially in problematic cases. However, traffic regulators did outperform object placement regarding pedestrian speeds. In addition, route guidance indicate little effects, and did not improve neither the speed nor density values under most cases tested. That said, route guidance did outperform the other crowd management measures on delivering stable speed outcomes under extremely problematic conditions.

While results clearly indicated the potential of in-event crowd management measures, one should always nourish the possible adverse effects, and thus consider the “do nothing”, or zero case, option (Haan & Heer, 2015, p. 93). In addition, Karbovskii et al. (2019) already pointed to similar occurrences of adverse effects with the application of crowd management measures. In this research, it was found that all in-event crowd management measures do have their limitations. While often outperforming the zero case, all measures also showed to worsen outcomes compared to the zero case under at least some of the tested conditions. It will therefore always be up to the specific desires of the decision-maker as to what in-event crowd management measure would be best, given the circumstance, where the zero case should always be included into the equation.

6.2. Strengths, limitations and future research

As a strength, this study presented three novel steps for model-based experimentation, based on techniques from exploratory modelling and analysis (Bankes, 1993), in the field of in-event crowd management: (1) exploration; (2) selection; and (3) evaluation. By combining microscopic pedestrian simulation framework Vadere (Kleinmeier et al., 2019), with the exploratory capabilities of the EMA Workbench (Kwakkel, 2017), this work synergizes an extensive and open-source pedestrian simulator with one of the most extensive Python based exploratory modelling tool collections. Not only did the application of the case study result in a comprehensive agent-based model of the Grote Markt in Breda, The Netherlands. It, most of all, demonstrated the potential of the proposed methodology in offering operational support for in-event crowd management.

Every study has its limitations. One of the limitations in this study has regard to the potential effect of model specific data gathering implementation on found results. Specific to the case of the Grote Markt, the placement of the four density-based measurement areas, the averaging of all the speed values, and the lack of warm-up time, are expected to have influenced the results found. If other measurement areas would be chosen with a different calculation of density and speed values, results could for instance capture high density clusters of pedestrians that were not included currently. However, given the model specific implementations and assumptions, this work does point to the promising results of traffic regulators on the Grote Markt.

Another limitation has regard to the underlying model assumptions. Three model-based assumption had to be made regarding limited data, which increased the complexity of verifying, calibrating and validating the constructed agent-based model (Crooks et al., 2008). First, origin and destinations of pedestrians had to be simplified, thereby neglecting shopping behavior such as pedestrians pausing in front of stores. This shopping behavior could make a difference in the real world (Hahm et al., 2019). Second, pedestrians are randomly placed on the model source areas, and especially when placed and clustered upon spawning in groups, this could affect dynamics. For instance, when more groups are present in the model, pedestrians will be placed closer together from the start of the simulation, thereby affecting the measured density values. Third, layout elements of the Grote Markt, for instance the location of the terraces and the addition of elements like street lanterns or fountains, are factors that are expected to influence crowd dynamics. In this work, these elements were based on Open Street Map data and satellite images, but do lack details like the presence of street lanterns or the exact pedestrians routes over the terraces.

However, note that assumptions specifically related to model underlying pedestrian behavior mechanism, such as the locomotion and group model, did not have to be made in this work. Those assumptions were based on previous work done with, and thereby embedded in, the Vadere simulation framework (Kleinmeier et al., 2019; Köster et al., 2011; M. J. Seitz & Köster, 2012; Von Sivers & Köster, 2015).

The last limitation relates to computational challenges. The proposed methodology is aimed to handle this in a novel way, making it possible to quickly search through a large uncertainty space. However, computational requirements remain high. This research is performed in an academic setting and in collaboration with the research department of Geodan. Therefore, computational resources were available that would most likely not be available for all, especially in the more practical field of crowd management. Furthermore, the used optimal step model in Vadere is currently not yet ready to simulate an entire city with thousands or more pedestrians. Nonetheless, computational hardware gets better every year, and many academic and private organizations have more computational resources to their availability. Therefore, with practical guidance, this research believes in the feasibility of the proposed methodology, even outside of the academic context.

Considering the above-mentioned limitations, one apparent focus of future research could be on the improvement of the simulation techniques specific to the underling model. Throughout more extensive validation studies, specifically targeted at the Grote Markt, and the integration of more urban data, the model could potentially be extended significantly. Moreover, the model could be extended spatially, enlarging the focus area. The optimal step implementation in Vadere would, however, need to be more efficient to allow the simulation of a larger number of agents at a time. Alternatively, other models of locomotion, such as Cellular Automata, could propose solutions in terms of computational requirements (Burstedde et al., 2001; Kleinmeier et al., 2019), but would, in the opinion of this research, lead to less accurate results. The question, however, remains how much this accuracy would matter when the area of focus is enlarged significantly.

Other future research plans are on extending on the pedestrian dynamics mechanisms currently supported in Vadere. While Vadere does support a wide range of extensive mechanisms, such as readily implemented locomotion models and a group model, further research could broader current implementations. Group models could potentially be included for the other models of locomotion, enabling better comparative behavior between the models. Alternatively, model specific elements such as warm-up time support could be added. Note that further development on Vadere is encouraged by its authors, and made easy by design (Kleinmeier et al., 2019).

Moreover, this work stresses the optimization of computational requirements needed. The proposed methodology of this work requires significant computational power, which is often the limiting factor in performing the experiments, even when powerful resources are present. Further work could build on the proposed methodology, to reduce computational requirements more. In addition, while the constructed EMA Workbench connector of this work already reduced computational time by the introduction of parallel processing, it could potentially be more efficient. Currently, each new model run is initiated by a new subprocess instance, calling the Vadere console executive. In the future, this could potentially be optimized by persisting the connection to the Vadere simulator, and passing new case properties directly.

As a last note on future research, practical support is needed to assist in-event crowd management in the application of the proposed methodology. While the proposed methodological steps of this work were applied successfully, by both keeping computational requirements manageable and by handling inherent complexity related to crowd modeling, support can be extended by making it easy for crowd managers to apply the methodology. This support could be shaped, for instance, by a detailed instruction on how to set up model runs, or how to utilize powerful, cloud based, computing. This work was performed in collaboration with Geodan, and thereby aimed to support usage of the proposed methodology outside of the academic field already by demonstrating its potential directly inside the practical field.

6.3. Theoretical & practical implications

The theoretical implications of this work are twofold. First, and arguably the most significant, this work provided the foundation on how to utilize crowd models for operational support on in-event crowd management. Secondly, the application on the Grote Markt case study showcased extensive insights into the effects of in-event crowd management measures on pedestrian speed and density values. This illustrated the potential of the traffic regulator measures to potentially mitigate unsafe situations under a large number of circumstances.

The case study on the Grote Markt also directly leads to three practical implications of this work concerning in-event crowd management. First, following the initial presented results, decision-makers responsible for crowd management in inner-city areas could successfully apply in-event crowd management measures to increase pedestrian speeds and decrease densities in problematic areas. Second, implementing in-event crowd management measures should be handled with care, as results did indicated decreased performance under some circumstances compared to doing nothing. Third, the proposed methodology accounts for the variation of effects under different circumstances by evaluating control measures over three sets—problematic, good, and worst—of circumstances. Crowd managers could, by utilizing the proposed steps of this work, use the insights from the model exploration in order to select diverse sets of circumstances to analyze their measures.

Overall, this study points to the potential for decision-makers to utilize traffic regulators in areas with dense terraces, in order to completely block problematic areas. This mitigates direct safety concerns, and was found to only have limited adverse effects. That said, attention should always be given to the creation of new, problematic, high density clusters, as a direct result of utilizing traffic regulators. Most of all, with the right practical guidance, the methodology of this work can be applied to more operational crowd management cases, and significantly aid crowd managers in various sectors.

6.4. Conclusion

Crowd management is a crucial element in keeping situations safe. Models can help understand in-event crowd management system more thoroughly, and illustrate potential effects of measures before they have to be implemented in real life. However, applications of crowd models for operational support on in-event crowd management are sparse. This study presented three methodological steps—exploration, selection, and evaluation—to utilize crowd models for operational crowd management. These steps were then applied to the case study of the Grote Markt, in the city of Breda, The Netherlands. The research question was: *"What effect do the in-event crowd management measures—traffic regulators, directional guidance, and object placement—have on the density and walking speed of* *pedestrians in the Grote Markt, Breda?"*. This question was answered by: (1) building a microscopic crowd model in Vadere, thereby enabling rapid yet sophisticated model development; and (2) utilizing this model, based on the proposed steps for experimentation, using techniques from the field of exploratory modelling and analysis. A connector between Vadere and the Python based Exploratory Modelling and Analysis (EMA) Workbench was constructed to synergize these two steps. Results illustrated the importance of the number of pedestrians origination from the Veemarktstraat and the amount of personal space pedestrians strive for. It was found that the proposed traffic regulators measure performed best on the Grote Markt. Not only did results indicate robust improvements of this measure on the pedestrian speed and density, the measure could also prevent the safety density threshold from being reached most out of all tested measures. Limitations of this work are that, for the application of the proposed methodology, results will always depend on the underlying modelling choices. In the focus case, examples of this are the way in which data is gathered, with specific measurement areas for the densities, and model specific implementations of spawn locations and initial densities of pedestrians. In addition, support for the practical usage of the presented model-based steps could be extended. In conclusion, this work presented a methodology to deal with the many uncertainties in the field of crowd simulations, and managed to mind computational limitations by presenting a way to quickly search through a wide range of alternative circumstances. With the proposal of the methodological steps, this work provides the needed stepping stone for operational support on crowd management. One that utilizes crowd models to understand in-event crowd management systems, and thereby enables the comparison between different in-event measures before they have to be implemented in real life.

References

7x zomerse terrassen in breda! [Stappen & shoppen breda]. (2018, November 4). Retrieved March 8, 2022, from https://www.stappen-shoppen.nl/nieuws/de-leukste-terrassen-per-gebied-in-breda-5ac375d32a5ab0606809e4c4

Argaleo. (2021, December 7). *Brabant consortium wins EU project for smart city dashboard* [Argaleo]. Retrieved April 26, 2022, from https://www.argaleo.com/en/2021/12/07/brabants-consortium-wint-eu-project-voor-smart-city-dashboard/

Asimakopoulou, E., & Bessis, N. (2011). Buildings and crowds: Forming smart cities for more effective disaster management. *2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 229–234. https://doi.org/10.1109/IMIS.2011.129

Autoriteit Persoonsgegevens. (2021, July 30). *AP publiceert aanbevelingen voor smart cities*. Retrieved December 15, 2021, from https://autoriteitpersoonsgegevens.nl/nl/nieuws/ap-publiceert-aanbevelingen-voor-smart-cities#subtopic-1733

Bankes, S. (1993). Exploratory modeling for policy analysis [Publisher: INFORMS]. *Operations Research*, *41*(3), 435–449. Retrieved March 9, 2022, from https://www.jstor.org/stable/171847

Burstedde, C., Klauck, K., Schadschneider, A., & Zittartz, J. (2001). Simulation of pedestrian dynamics using a two-dimensional cellular automaton. *Physica A: Statistical Mechanics and its Applications*, *295*(3), 507–525. https://doi.org/10.1016/S0378-4371(01)00141-8

Crooks, A., Castle, C., & Batty, M. (2008). Key challenges in agent-based modelling for geo-spatial simulation. *Computers, Environment and Urban Systems*, *32*(6), 417–430. https://doi.org/10.1016/j.compenvurbsys.2008.09.004

Duives, D. C., Daamen, W., & Hoogendoorn, S. P. (2013). State-of-the-art crowd motion simulation models. *Transportation Research Part C: Emerging Technologies*, *37*, 193–209. https://doi.org/10.1016/j.trc.2013.02.005

Duives, D. C., Daamen, W., & Hoogendoorn, S. P. (2015). Quantification of the level of crowdedness for pedestrian movements. *Physica A: Statistical Mechanics and its Applications*, *427*, 162–180. https://doi.org/10.1016/j.physa.2014.11.054

Echeverría-Huarte, I., Garcimartín, A., Hidalgo, R. C., Martín-Gómez, C., & Zuriguel, I. (2021). Estimating density limits for walking pedestrians keeping a safe interpersonal distancing. *Scientific Reports*, *11*(1), 1534. https://doi.org/10.1038/s41598-020-79454-0

Foramitti, J. (2021). AgentPy: A package for agent-based modeling in python. *Journal of Open Source Software*, *6*(62), 3065. https://doi.org/10.21105/joss.03065

Garcia-Retuerta, D., Chamoso, P., Hernández, G., Guzmán, A. S. R., Yigitcanlar, T., & Corchado, J. M. (2021). An efficient management platform for developing smart cities: Solution for real-time and future crowd detection. *Electronics*, *10*(7), 765. https://doi.org/10.3390/electronics10070765

George, G., Howard-Grenville, J., Joshi, A., & Tihanyi, L. (2016). Understanding and tackling societal grand challenges through management research. *Academy of Management Journal*, *59*(6), 1880–1895. https://doi.org/10.5465/amj.2016.4007

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*(1), 3–42. https://doi.org/10.1007/s10994-006-6226-1

Gödel, M. (2022, January 25). *Systematic parameter analysis to reduce uncertainty in crowd simulations* (Doctoral dissertation). Technische Universität München. Retrieved March 30, 2022, from https://mediatum.ub.tum.de/doc/1633090/1633090.pdf

Gorrini, A., Shimura, K., Bandini, S., Ohtsuka, K., & Nishinari, K. (2014). Experimental investigation of pedestrian personal space toward modeling and simulation of pedestrian crowd dynamics. *Transportation Research Record*, (2421), 57–63. https://doi.org/10.3141/2421-07

Grassmann, W. K. (2009). When, and when not to use warm-up periods in discrete event simulation. *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, 1–6. https://doi.org/10.4108/ICST.SIMUTOOLS2009.5603

Groeneweg, J. (2021, November 23). *Vanaf morgen 1,5 meter afstand houden weer verplicht, boete is 95 euro*. Retrieved January 11, 2022, from https://nos.nl/l/2406793

Haan, A. d., & Heer, P. d. (2015). *Solving complex problems: Professional group decision-making support in highly complex situations* (Second edition). Eleven International Publishing.

Haciomeroglu, M., Laycock, R. G., & Day, A. M. (2008). Automatic spatial analysis and pedestrian flow control for real-time crowd simulation in an urban environment. *The Visual Computer*, *24*(10), 889–899. https://doi.org/10.1007/s00371-008-0291-3

Hahm, Y., Yoon, H., & Choi, Y. (2019). The effect of built environments on the walking and shopping behaviors of pedestrians; a study with GPS experiment in sinchon retail district in seoul, south korea. *Cities*, *89*, 1–13. https://doi.org/10.1016/j.cities.2019.01.020

Halim, R. A., Kwakkel, J. H., & Tavasszy, L. A. (2016). A scenario discovery study of the impact of uncertainties in the global container transport system on european ports. *Futures*, *81*, 148–160. https://doi.org/10.1016/j.futures.2015.09.004

Hamarat, C., Kwakkel, J. H., Pruyt, E., & Loonen, E. T. (2014). An exploratory approach for adaptive policymaking by using multi-objective robust optimization. *Simulation Modelling Practice and Theory*, *46*, 25–39. https://doi.org/10.1016/j.simpat.2014.02.008

Hegemann, J. (2022, July 21). *Why aren't more researchers using open source?* Retrieved February 10, 2022, from https://opensource.com/article/17/6/open-source-research

Helbing, D., & Molnar, P. (1998). Social force model for pedestrian dynamics. *Physical Review E*, *51*. https://doi.org/10.1103/PhysRevE.51.4282

Helton, J. C., & Davis, F. J. (2003). Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliability Engineering & System Safety*, *81*(1), 23–69. https://doi.org/10.1016/S0951-8320(03)00058-9

Higgins, B. (n.d.). *Astroworld festival tragedy spotlights best practices in crowd management*. Retrieved November 11, 2021, from https://www.usatoday.com/story/opinion/2021/11/10/astroworld-tragedy-puts-spotlight-best-practices-crowd-management/6362061001/

Johansson, A., Helbing, D., & Shukla, P. K. (2007). Specification of the social force pedestrian model by evolutionary adjustment to video tracking data. *Advances in Complex Systems*, *10*, 271–288. https://doi.org/10.1142/S0219525907001355

Kaiser, M. S., Lwin, K. T., Mahmud, M., Hajializadeh, D., Chaipimonplin, T., Sarhan, A., & Hossain, M. A. (2018). Advances in crowd analysis for urban applications through urban event detection. *Ieee Transactions on Intelligent Transportation Systems*, *19*(10), 3092–3112. https://doi.org/10.1109/TITS.2017.2771746

Kaminka, G. A., & Fridman, N. (2018). Simulating urban pedestrian crowds of different cultures. *Acm Transactions on Intelligent Systems and Technology*, *9*(3), 27. https://doi.org/10.1145/3102302

Karbovskii, V., Severiukhina, O., Derevitskii, I., Voloshin, D., Presbitero, A., & Lees, M. (2019). The impact of different obstacles on crowd dynamics. *Journal of Computational Science*, *36*, 100893. https://doi.org/10.1016/j.jocs.2018.06.010

Kazil, J., Masad, D., & Crooks, A. (2020). Utilizing python for agent-based modeling: The mesa framework. In R. Thomson, H. Bisgin, C. Dancy, A. Hyder, & M. Hussain (Eds.), *Social, cultural, and behavioral modeling* (pp. 308–317). Springer International Publishing. https://doi.org/10.1007/978-3-030-61255-9_30

Kleinmeier, B., Zönnchen, B., Gödel, M., & Köster, G. (2019). Vadere: An open-source simulation framework to promote interdisciplinary understanding. *Collective Dynamics*, *4*. https://doi.org/10.17815/CD.2019.21

Köster, G., Seitz, M., Treml, F., Hartmann, D., & Klein, W. (2011). On modelling the influence of group formations in a crowd. *Contemporary Social Science*, *6*(3), 397–414. https://doi.org/10.1080/21582041.2011.619867

Köster, G., Treml, F., Seitz, M., & Klein, W. (2014). Validation of crowd models including social groups. In U. Weidmann, U. Kirsch, & M. Schreckenberg (Eds.), *Pedestrian and evacuation dynamics 2012* (pp. 1051–1063). Springer International Publishing. Retrieved June 1, 2022, from https://doi.org/10.1007/978-3-319-02447-9_87

Koukopoulos, Z., Koukopoulos, D., & Jung, J. J. (2018). Real-time crowd management for cultural heritage events: A case study on carnival parades. *Journal of Ambient Intelligence and Smart Environments*, *10*(3), 275–287. https://doi.org/10.3233/AIS-180485

Kwakkel, J. H. (2017). The exploratory modeling workbench: An open source toolkit for exploratory modeling, scenario discovery, and (multi-objective) robust decision making. *Environmental Modelling & Software*, *96*, 239–250. https://doi.org/10.1016/j.envsoft.2017.06.054

Kwakkel, J. H., & Jaxa-Rozen, M. (2016). Improving scenario discovery for handling heterogeneous uncertainties and multinomial classified outcomes. *Environmental Modelling & Software*, *79*, 311–321. https://doi.org/10.1016/j.envsoft.2015.11.020

Lu, L., Chan, C.-Y., Wang, J., & Wang, W. (2017). A study of pedestrian group behaviors in crowd evacuation based on an extended floor field cellular automaton model. *Transportation Research Part C: Emerging Technologies*, *81*, 317–329. https://doi.org/10.1016/j.trc.2016.08.018

Lubaś, R., Wąs, J., & Porzycki, J. (2016). Cellular automata as the basis of effective and realistic agent-based models of crowd behavior. *The Journal of Supercomputing*, *72*(6), 2170–2196. https://doi.org/10.1007/s11227-016-1718-7

Mayr, C., & Köster, G. (2021). Social distancing with the optimal steps model. *Collective Dynamics*, *6*. https://doi.org/10.17815/CD.2021.116

Moss, S., & Edmonds, B. (2005). Sociology and simulation: Statistical and qualitative cross-validation. *American Journal of Sociology*, *110*(4), 1095–1131. https://doi.org/10.1086/427320

Moussaid, M., Helbing, D., Garnier, S., Johansson, A., Combe, M., & Theraulaz, G. (2009). Experimental study of the behavioural mechanisms underlying self-organization in human crowds. *Proceedings of the Royal Society B-Biological Sciences*, *276*(1668), 2755–2762. https://doi.org/10.1098/rspb.2009.0405

Moussaid, M., Helbing, D., & Theraulaz, G. (2011). How simple rules determine pedestrian behavior and crowd disasters. *Proceedings of the National Academy of Sciences of the United States of America*, *108*(17), 6884–6888. https://doi.org/10.1073/pnas.1016507108

Moussaïd, M., Perozo, N., Garnier, S., Helbing, D., & Theraulaz, G. (2010). The walking behaviour of pedestrian social groups and its impact on crowd dynamics. *PLoS ONE*, *5*(4). https://doi.org/10.1371/journal.pone.0010047

Netherlands Space Office. (n.d.). *Kaart* [Satellietdataportaal]. Retrieved March 31, 2022, from https://www.satellietdataportaal.nl/

NOS. (2021, January 5). *Drukke winkelstraten na versoepelingen, veel Duitsers in grenssteden*. Retrieved March 15, 2022, from https://nos.nl/l/2378983

Omroep Flevoland. (2022, February 17). *Politie zoekt dader mishandeling in drukke winkelstraat op 5 december*. Retrieved April 20, 2022, from https://www.omroepflevoland.nl/nieuws/273665/politie-zoekt-dader-mishandeling-in-drukke-winkelstraat-op-5-december

Park, A. J., Zouaghi, H., & Tsang, H. H. (2018). Crowd control strategy framework using real-time 3d simulations. In S. Chakrabarti & H. N. Saha (Eds.), *2018 IEEE 9th annual information technology, electronics and mobile communication conference (IEMCON)* (pp. 253–258). Ieee. https://doi.org/10.1109/IEMCON.2018.8614808

Pelechano, N., Allbeck, J. M., & Badler, N. I. (2007). Controlling individual agents in high-density crowd simulation. *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 99–108. https://repository.upenn.edu/hms/210

Pellegrini, S., Ess, A., Schindler, K., & van Gool, L. (2009). You'll never walk alone: Modeling social behavior for multi-target tracking. *2009 IEEE 12th International Conference on Computer Vision*, 261–268. https://doi.org/10.1109/ICCV.2009.5459260

RiMEA e.V. (2016, October 3). *Richtlinie für mikroskopische entfluchtungsanalysen*. RiMEA e.V. Retrieved March 17, 2022, from https://rimea.de/de/rimea-projekt/richtlinie-fuer-mikroskopische-entfluchtungsanalysen/

Sargent, R. G. (2010). Verification and validation of simulation models. *Proceedings of the 2010 Winter Simulation Conference*, 166–183. https://doi.org/10.1109/WSC.2010.5679166

Schaap, S., & Dijkshoorn, M. (2020, May). *Reguleren van drukte in de openbare ruimte: Handreiking crowd management bij het verantwoord openstellen*. VNG. https://vng.nl/sites/default/files/2020-06/vng-handreiking-crowmanagement.pdf

Schriber, T. J., Brunner, D. T., & Smith, J. S. (2015). Inside discrete-event simulation software: How it works and why it matters. *2015 Winter Simulation Conference (WSC)*, 1–15. https://doi.org/10.1109/WSC.2015.7408149

Seitz, M., Köster, G., & Pfaffinger, A. (2014). Pedestrian group behavior in a cellular automaton. In U. Weidmann, U. Kirsch, & M. Schreckenberg (Eds.), *Pedestrian and evacuation dynamics 2012* (pp. 807–814). Springer International Publishing. Retrieved April 6, 2022, from https://doi.org/10.1007/978-3-319-02447-9_67

Seitz, M. J., & Köster, G. (2012). Natural discretization of pedestrian movement in continuous space. *Physical Review E*, *86*(4), 046108. https://doi.org/10.1103/PhysRevE.86.046108

Seitz, M. J., & Köster, G. (2014). How update schemes influence crowd simulations. *Journal of Statistical Mechanics: Theory and Experiment*, *2014*(7), P07002. https://doi.org/10.1088/1742-5468/2014/07/P07002

Sharma, D., Bhondekar, A. P., Shukla, A. K., & Ghanshyam, C. (2018). A review on technological advancements in crowd management. *Journal of Ambient Intelligence and Humanized Computing*, *9*(3), 485–495. https://doi.org/10.1007/s12652-016-0432-x

Shende, A., Singh, M. P., & Kachroo, P. (2011). Optimization-based feedback control for pedestrian evacuation from an exit corridor. *IEEE Transactions on Intelligent Transportation Systems*, *12*(4), 1167–1176. https://doi.org/10.1109/TITS.2011.2146251

Sobol', I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and Computers in Simulation*, *55*(1), 271–280. https://doi.org/10.1016/S0378-4754(00)00270-6

Speakman, K. (2021, March 12). *Nearly 300 astroworld lawsuits to be combined into single case* [Forbes]. Retrieved March 8, 2022, from https://www.forbes.com/sites/kimberleespeakman/2021/12/03/nearly-300-astroworld-lawsuits-to-be-combined-into-single-case/

Suganya, P., Gaurav, T., Rampuriya, P., & Dey, S. (2019). Locating object - of - interest & preventing stampede using crowd management. *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, 243–247. https://doi.org/10.1109/ICCMC.2019.8819812

Sutheerakul, C., Kronprasert, N., Kaewmoracharoen, M., & Pichayapan, P. (2017). Application of unmanned aerial vehicles to pedestrian traffic monitoring and management for shopping streets. *Transportation Research Procedia*, *25*, 1717–1734. https://doi.org/10.1016/j.trpro.2017.05.131

Tao, W. (2013). Interdisciplinary urban GIS for smart cities: Advancements and opportunities. *Geo-Spatial Information Science*, *16*(1), 25–34. https://doi.org/10.1080/10095020.2013.774108

Teo, J. S. E., Schmoecker, J.-D., Leon, F., Li, J. Y.-T., Ji, J., Atanasiu, G., & Taniguchi, E. (2015). Agent-based evacuation model considering field effects and government advice. *Transportation Research Record*, *2532*(1), 129–140. https://doi.org/10.3141/2532-15

The Guardian. (2021). Astroworld: Deaths of 10 people at houston concert ruled accidental. *The Guardian*. Retrieved March 8, 2022, from https://www.theguardian.com/music/2021/dec/16/astroworld-festival-deaths-ruled-accidental

Van Dam, K. H., Nikolic, I., & Lukszo, Z. (2013). *Agent-based modelling of socio-technical systems*. Springer Science+Business Media. https://doi.org/10.1007/978-94-007-4933-7

Vermeir, A., & Bersini, H. (2015). Best practices in programming agent-based models in economics and finance. In F. Amblard, F. J. Miguel, A. Blanchet, & B. Gaudou (Eds.), *Advances in artificial economics* (pp. 57–68). Springer International Publishing. https://doi.org/10.1007/978-3-319-09578-3_5

Von Sivers, I., Templeton, A., Künzner, F., Köster, G., Drury, J., Philippides, A., Neckel, T., & Bungartz, H.-J. (2016). Modelling social identification and helping in evacuation simulation. *Safety Science*, *89*, 288–300. https://doi.org/10.1016/j.ssci.2016.07.001

Von Sivers, I., & Köster, G. (2015). Dynamic stride length adaptation according to utility and personal space. *Transportation Research Part B: Methodological*, *74*, 104–117. https://doi.org/10.1016/j.trb.2015.01.009

Wąs, J., & Lubaś, R. (2014). Towards realistic and effective agent-based models of crowd dynamics. *Neurocomputing*, *146*, 199–209. https://doi.org/10.1016/j.neucom.2014.04.057

Weidmann, U. (1992). *Transporttechnik der fussgänger: Transporttechnische eigenschaften des fussgängerverkehrs, literaturauswertung*. ETH Zurich. Retrieved March 18, 2022, from http://hdl.handle.net/20.500.11850/47999

Wijermans, N., Conrado, C., van Steen, M., Martella, C., & Li, J. (2016). A landscape of crowd-management support: An integrative approach. *Safety Science*, *86*, 142–164. https://doi.org/10.1016/j.ssci.2016.02.027

Wilensky, U. (1999). *NetLogo*. Center for Connected Learning and Computer-Based Modeling. Northwestern University, Evanston, IL. http://ccl.northwestern.edu/netlogo/

Wilensky, U., & Rand, W. (2015). *An introduction to agent-based modeling: Modeling natural, social, and engineered complex systems with NetLogo*. The MIT Press.

Yu, Q., Hu, L., Alzahrani, B., Baranawi, A., Alhindi, A., & Chen, M. (2021). Intelligent visual-IoT-enabled real-time 3d visualization for autonomous crowd management. *IEEE Wireless Communications*, *28*(4), 34–41. https://doi.org/10.1109/MWC.021.2000497

Yuan, Y., Daamen, W., Duives, D., & Hoogendoorn, S. (2016). Comparison of three algorithms for real-time pedestrian state estimation - supporting a monitoring dashboard for large-scale events. *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2601–2606. https://doi.org/10.1109/ITSC.2016.7795974

Zhang, J., Klingsch, W., Schadschneider, A., & Seyfried, A. (2011). Transitions in pedestrian fundamental diagrams of straight corridors and t-junctions [Publisher: IOP Publishing]. *Journal of Statistical Mechanics: Theory and Experiment*, *2011*(6), P06004. https://doi.org/10.1088/1742-5468/2011/06/P06004

Zhang, J., Zheng, Y., Qi, D., Li, R., Yi, X., & Li, T. (2018). Predicting citywide crowd flows using deep spatio-temporal residual networks. *Artificial Intelligence*, *259*, 147–166. https://doi.org/10.1016/j.artint.2018.03.002

Zhou, H., Zheng, Z., Cen, X., Huang, Z., & Wang, P. (2021). A data-driven urban metro management approach for crowd density control. *Journal of Advanced Transportation*, *2021*, e6675605. https://doi.org/10.1155/2021/6675605

Zönnchen, B., & Köster, G. (2019). A parallel generator for sparse unstructured meshes to solve the eikonal equation. *Journal of Computational Science*, *32*, 141–147. https://doi.org/10.1016/j.jocs.2018.09.009

A

Vadere model details

This chapter presents additional details regarding the implementation of the pedestrian crowd model of the Grote Markt (Breda) in the Vadere modelling framework. This chapter can be read standalone, but it is advised to consult Chapter 3 first, regarding additional details of modelling components. A full copy of the model can be found on GitHub[1].

A Vadere model can be separated into six different components, which are subsequently showed when opening the to this research accompanied model file ("vadere.project"), or one of the specific Vadere scenario files, in the Vadere GUI. A Vadere scenario file represents a specific version of the model (e.g., the base case one and one with traffic regulators in place). Opening the accompanied overall model file automatically loads all Vadere scenario files. Together, the six model components form all the details needed for the Vadere simulator to perform a run. Specifics for each component can, however, variate per scenario. The six components are: (1) simulation; (2) topography, (3) model; (4) psychology; (5) perception; and (6) data output. This chapter discusses all six elements in the context of the constructed model of the Grote Markt.

A.1. Simulation settings

The simulation settings represent the overall model runtime settings. The simulation settings used in this research are illustrated in Table A.1, and are kept constant for all scenario files, with the only exception being the ones used for verification purposes. In the verification simulation settings, the simulation seed is set to a constant to keep full and individual reproducibility.

[1] https://github.com/floristevito/CrowdSim

Table A.1: Simulation settings

Setting	Value	Value for verification*
finishTime	100	100
simTimeStepLength	0.4	0.4
realTimeSimTimeRatio	0.0	0.0
writeSimulationData	true	true
visualizationEnabled	true	true
printFPS	false	false
digitsPerCoordinate	2	2
useFixedSeed	false	true
fixedSeed	n/a	-5190782782724114477
simulationSeed	n/a	-5190782782724114477

*Verification model files are targeted at full reproducibility.

A.2. Topography settings

The topography settings specify the spatial layout of the model, with the exact location of all physical model components. When using the Vadere GUI, a topography creator that can aid with topological model development can be used. The topological specification furthermore follows a JSON based structure, making it possible to generate model sections with other programming languages and inserting them into the topography settings. This work used a combination of the GUI topography creator and Python scripts in order to generate the overall model topography.

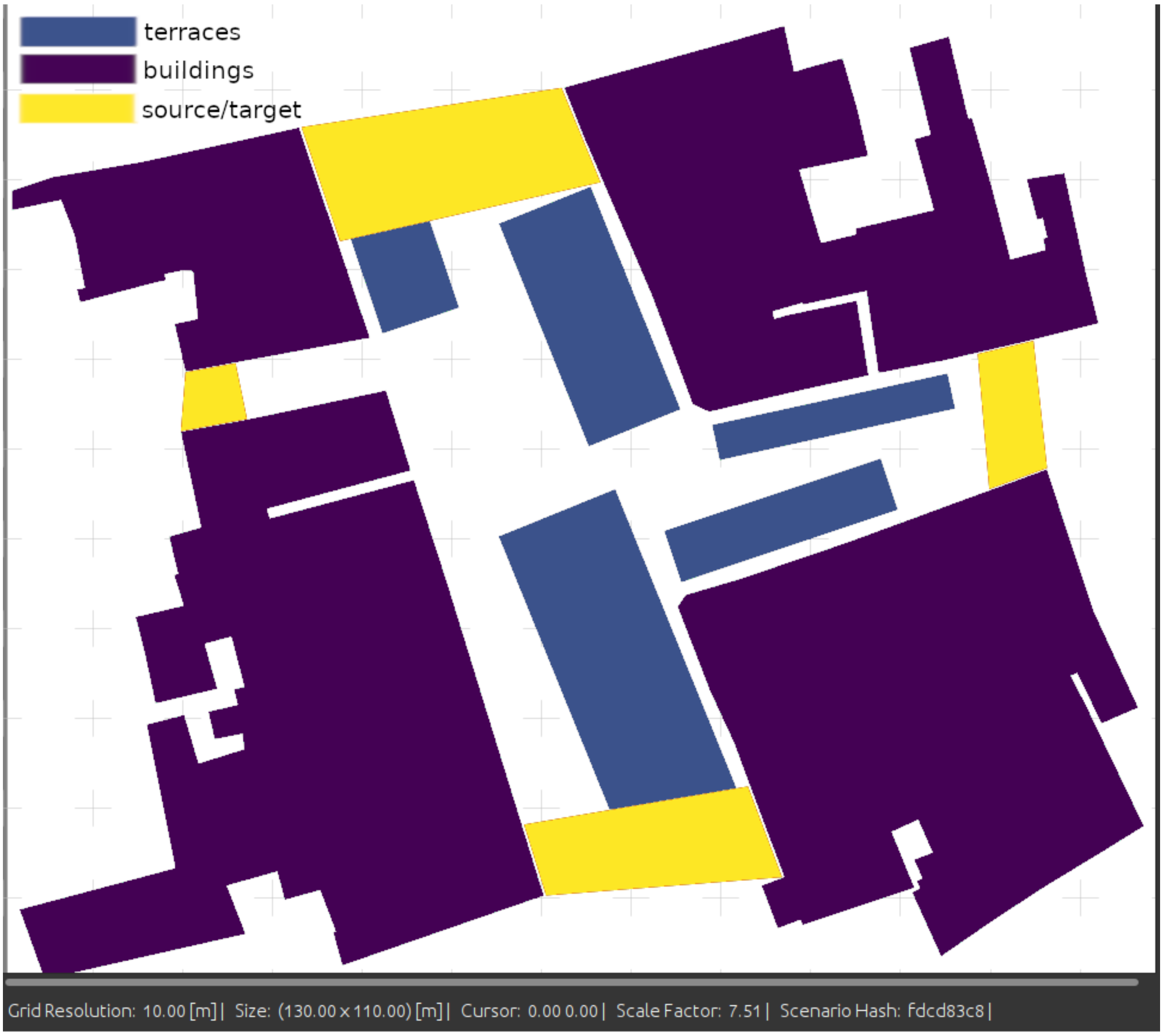


Figure A.1: Model topography

A.2.1. Base topography

The overall topography of the Grote Markt is illustrated in Figure A.1. Note that the topography is in scale to represent the Grote Markt realistically. However, exporting the model layout is realized with model screenshots, which are subsequently rescaled and formatted. In the original Vadere model, the used grid is in scale of 10 by 10 meter, and can be pointed out in the figure by the grid-based gray crosses in the background.

The terraces are generated with Python scrips, manually referenced by satellite imagery (Netherlands Space Office, n.d.). By manually referencing coordinates in the Vadere simulator to points in the imagery, the rough location of terraces could be inferred. The points then served as input to Python scripts, which can be found on the accompanied GitHub page, including documentation. The GitHub page additionally includes the full specification of the exact location of each of the terraces, translated into Vadere components.

Building elements are generated using data from Open Street Map [2]. The original Vadere repository provides a tool to convert Open Street Map data to Vadere topography objects. This script is slightly modified in this research, in order to directly store the output contents to files. The script, with usage instructions, input Open Street Map data, and output in the form of generated Vadere buildings with exact coordinates, are provided on GitHub.

Source and target elements represent the origin and destinations of pedestrians in the model. These objects are drawn in by hand using the Vadere GUI, and do not represent actual physical topographical objects. The exact coordinates of these objects can therefore be found in the full model file, included on GitHub.

A.2.2. Additional topography

Besides the base topographical elements illustrated in Figure A.1, additional components are subsequently used for specific model versions. The object placement crowd management measure, for instance, required a separate model specification, with additional line-barrier objects. All these additional topographical elements are drawn in by hand using the Vadere GUI. For each specific model version, a separate Vadere scenario file specification is included in the model. Therefore, the exact specification of all these additionally and manually drawn elements can be found in the model scenario file specifications, included on GitHub. In addition, the location and implementation of elements related to crowd management measure are discussed and specified in Chapter 3.

A.3. Model

The model settings specify the working of the model itself, and how pedestrians move in the model. Vadere thereby provides several models by default, which can be inserted with the Vadere GUI.

The Optimal Step Model (M. J. Seitz & Köster, 2012) is used in this research, provided as a default template by the Vadere simulator. With the enabling of this models comes the automatic specification of a relatively large amount of parameter values. The working of this model is discussed in Section 2.1.2. In addition, the full model specification and implementation in Vadere can be found in the work of Kleinmeier et al. (2019). The included model files on GitHub also list all underlying parameters. These values are extensively calibrated on previous work by the Vadere authors. These parameters are therefore not modified in this work, with the only exception being the maximum step duration. This setting was set to 10 seconds after model verification (see Section 3.3.2).

In addition to the Optimal Step Model, the Centroid Group Model was used in this research, based on the work of Köster et al. (2011). This group model follows the implementation as discussed in Section 2.2.2 and summarized in Table 2.3. This model is again provided by default in the Vadere simulator, and this research did not change any of the underlying specifications. The implementation in Vadere of the model as described in Köster et al. (2011), together with the full parameter specification, can be found in the model specification included on GitHub.

A.4. Psychology

Vadere provides settings for a so-called psychology layer. These mechanisms provide an additional working to determine the behavior of pedestrians, before they are moved by the locomotion layer (Optimal Step Model in this work). This research utilizes one cognition model of interest here. This layer can be activated by specifying the name in the physiology settings, here referred to by “CooperativeCognitionModel”. This model is a default model provided by Vadere, targeted for use in simulations with a large number of counter flowing pedestrians, as is the case in this work. By using this mechanism, pedestrians that are opposed to each other in the simulation are specified to swap their places. This makes sure that pedestrians do not get stuck when encountering other pedestrians moving in the opposite direction.

A.5. Perception

The perception layer in Vadere can be used in combination with the psychology layer, in order to implement additional stimuli. For example, when a model is targeted at recreating a bomb evacuation,

[2] https://www.openstreetmap.org

this perception layer can be used to create the stimuli to represent such a treat in the simulation. This layer is not used in this work.

A.6. Data output

The data output section of the Vadere model specifies exactly what is gathered with each model run. This research uses two different sets of data output specifications. One set is targeted at post simulation visualization, thereby including detailed information such as the exact location of each pedestrian throughout the simulation. The other set is used for experimentation, and includes meaningful model statistics for data analysis. For each crowd management measure, a separate model specification is made for visualization and experimentation, and included on GitHub.

A.6.1. Data collector for visualization

Table A.2 illustrates all the data processors used for model runs targeted at visualization. All data processors are included by default in Vadere, and only have to be activated and pointed to the right output files. In addition, the GUI automatically recognizes the generated .traj file and uses this for post visualization directly from the GUI.

Table A.2: Data collectors used for visualization

Data processor	Vadere default?	Output files	Remarks
FootStepProcessor	Yes	postvis.traj	Used for visualization
FootStepTargetIDProcessor	Yes	postvis.traj	Used for visualization
PedestrianOverlapProcessor	Yes	overlaps.csv	Optional statistic
NumberOverlapsProcessor	Yes	overlapCount.txt	Optional statistic
FootStepGroupIDProcessor	Yes	postvis.traj	Used for visualization
FootSTepGroupSizeProcessor	Yes	postvis.traj	Used for visualization

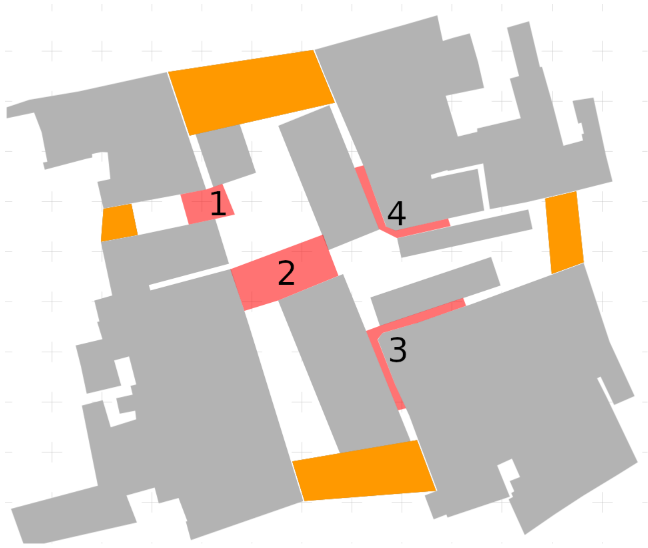


Figure A.2: Measurement areas in Vadere

Note. Red area = measurement area, gray area = obstacle (e.g. buildings and terraces), orange area = origin and destination points of pedestrians

A.6.2. Data collector for experimentation

The model is targeted at collecting statistics for pedestrian speed and density, as specified and discussed in Chapter 2 and 3. In order to collect this data in the Vadere model, measurement areas are needed, as discussed in detail in Section 3.3.1. An overview of these areas are illustrated in Figure

A.2. Note that the measurement area used for the speed data collection covers the entire model environment, and is therefore excluded from the figure. These areas are drawn by hand using the Vadere GUI. The exact specification of the areas can be found in the model specification, included on GitHub.

Table A.3 illustrates the data processors used for experimental runs. In additional to data collectors provided by default in Vadere, new collectors had to be constructed in order to gather scalar output values. These collectors are written in Java, and calculate the average and maximum from a sequence of speed and density values. The Vadere executable attached on GitHub includes these additional collectors. For the Java source code of these collectors, please refer to the forked and publicly hosted version of Vadere utilized in this research[3]. This fork includes all the Vadere source code additions made within this work, which entails the new data collections and the change made to the spawn frequency (see Section 3.3.1). Note that these changes are included on the “CrowdSim_thesis_floris_additions” branch.

Table A.3: Data collectors used for experimentation

Data processor	Vadere default?	Output files	Remarks
AreaSpeedProcessor	Yes	n/a	Used by mean speed processor
PedestrianPositionProcessor	Yes	n/a	Used by mean speed processor
PedestrianVelocityProcessor	Yes	n/a	Used by mean speed processor
MeanAreaSpeedProcessor	No	speed.txt	Measured over entire model area
AreaDensityCountingNormedProcessor*	Yes	n/a	Used by density processors
MeanAreaDensityCountingNormedProcessor*	No	density.txt	Measured in specific areas
MaxAreaDensityCountingNormedProcessor*	No	density.txt	Measured in specific areas

*Included four times, one for each density measurement area

[3] https://github.com/floristevito/vadere